

UNIVERSIDAD CARLOS III DE MADRID



INGENIERÍA TÉCNICA INFORMÁTICA DE GESTIÓN

PROYECTO FIN DE CARRERA

VISOR GRÁFICO DE REDES DE REFERENCIAS CIENTÍFICAS

Autor: Rubén Redondo Correas

Tutor: Tomás Eduardo De la Rosa Turbides

Leganés, Julio 2015

Agradecimientos

A Claudia y Pedro, mi hijos, y a mi esposa Loli por el apoyo y ánimo que me han dado, o desapocho en el caso de los pequeños con sus continuas interrupciones. Por que sin ellos no estaríamos en este punto de mi carrera y nunca hubiera terminado lo que hace tantos años empecé.

Aunque suene a tópico a Tomas, mi tutor, por su paciencia, disponibilidad y cercanía. Por ayudarme en todo momento y comprender las dificultades en mi vida. Por proponerme este proyecto y que haya sido capaz de realizarlo.

Y a todos los que me han enseñado lo que sé en el mundo de la informática: profesores, compañeros de clase, amigos, compañeros de trabajo y a todos aquellos que en mayor o menor medida han compartido su conocimiento conmigo.

Resumen

En la actualidad existe una amplia variedad de aplicaciones de gestión de referencia y buscadores de documentación académica. Estas aplicaciones aunque gestionan e informan de las referencias de los documentos, no permiten tener una visión global de las relaciones, por sus referencias, entre documentos.

Este proyecto busca suplir esta carencia y dar una representación gráfica a la relación por referencia entre documentos. Además de poder visualizar la importancia de un documento por el número de citas recibidas. Y así poder localizar de una forma más cómoda aquellos documentos desconocidos de interés para el usuario.

El proyecto ha finalizado con la creación de una aplicación web, donde a partir de una búsqueda en la web CiteSeerX, buscador de publicaciones académicas y científicas. Se genera una representación gráfica en forma de grafo con las relaciones entre documentos por sus referencias.

Abstract

Nowadays there is a wide variety of reference management applications and search academic documentation. Although these applications manage and report the references of the documents, do not give a global view of relations, by reference, between documents.

This project seeks to fill this void and gives a graphical representation of the reference relationships between documents. Besides, it should be able to show the relevance of a document by the number of citations received. Thus, the user will realize in a more comfortable way the importance of those unknown documents to the user.

The project ended with the creation of a web application that uses the web CiteSeerX as the search engine of academic and scientific publications. A graphical representation is generated in form of graph with relationships between documents by reference.

Índice General

ÍNDICE GENERAL	6
ÍNDICE DE ILUSTRACIONES	10
ÍNDICE DE TABLAS	12
1. INTRODUCCIÓN	15
1.1. INTRODUCCIÓN	15
1.2. OBJETIVOS	17
1.3. ORGANIZACIÓN DEL DOCUMENTO.....	18
2. ESTADO DEL ARTE	19
2.1. ESTUDIOS PREVIOS	19
2.1.1. GESTORES DE REFERENCIAS.....	19
2.1.1.1 MENDELEY	20
2.1.1.2 Otros Gestores de Referencia	21
2.1.2. BUSCADORES DE DOCUMENTOS ACADÉMICOS.....	22
2.1.2.1 CiteSeerX.....	22
2.1.2.2 Otros buscadores de Documentos Académicos	24
2.1.3 GRAFO SOCIAL	27
2.2. TECNOLOGÍAS APLICADAS	28
2.2.1. HTML	28
2.2.2. HOJA DE ESTILOS CSS	28
2.2.3. PHP.....	29
2.2.4. JAVASCRIPT	29

2.2.5. JQUERY	30
2.2.6. LIBRERÍA GRÁFICA VISJP.....	30
2.2.7. AJAX	31
3 ANÁLISIS, DISEÑO E IMPLEMENTACIÓN.....	33
3.1 ANÁLISIS.....	33
3.1.1 REQUISITOS USUARIOS	33
3.1.2 CASOS DE USO	39
3.1.2.1 Descripción gráfica.....	39
3.1.2.2 Descripción textual	40
3.1.3 MATRIZ DE TRAZABILIDAD	44
3.2 DISEÑO	45
3.2.1 ARQUITECTURA DE LA APLICACIÓN	45
3.2.2 DISEÑO DE LA APLICACIÓN WEB	46
3.2.2.1 Diagrama de Navegación	46
3.2.2.2 Plantillas Aplicación web	47
3.2.3 ESTRUCTURA DE DATOS.....	50
3.3 IMPLEMENTACIÓN	52
3.3.1 FICHEROS.....	54
3.3.1.1 Index.html.....	54
3.3.1.2 Grafo.html.....	54
3.3.1.3 ba-simple-proxy.php	54
3.3.1.4 GlobalValor.js	55
3.3.1.5 cite-network.js	55

3.3.1.6	DatosGrafo.js	56
3.3.1.7	Archivo.js.....	57
3.4	PRUEBAS	58
3.4.1	PRUEBA VISUALIZACIÓN	59
3.4.2	PRUEBAS DE REQUISITOS.....	59
3.4.3	PRUEBAS DE ESFUERZO	62
4	PLANIFICACIÓN	63
4.1	PLANIFICACIÓN.....	63
4.1.1	MODELO DEL CICLO DE VIDA.....	63
4.1.2	DIAGRAMA DE GANTT	65
4.2	PRESUPUESTO.....	67
4.2.1	RESUMEN DE HORAS DEDICADAS.....	67
4.2.2	RESUMEN DE PERSONAL.....	68
4.2.3	RESUMEN DE HARDWARE	69
4.2.4	RESUMEN DE SOFTWARE Y LICENCIAS.....	70
4.2.5	RESUMEN DE MATERIAL FUNGIBLE.....	71
4.2.6	RESUMEN DEL PRESUPUESTO TOTAL.....	72
5	CONCLUSIONES Y TRABAJOS FUTUROS.....	73
5.1	CONCLUSIONES	73
5.2	TRABAJOS FUTUROS	74
6	ANEXOS.....	75
•	MANUAL DE USUARIO.....	75
•	REFERENCIAS	80

• ACRÓNIMOS.....	81
------------------	----

Índice de Ilustraciones

Ilustración 1: Resultado Búsqueda en CiteSeerX.....	23
Ilustración 2: Ficha de Documento en CiteSeerX.....	23
Ilustración 3: Ejemplo Grafico de MAS.....	26
Ilustración 4: Ejemplo de un Grafo Social	27
Ilustración 5: Comparativa Ajax-Tradicional	32
Ilustración 6: Caso de Uso Buscador	39
Ilustración 7: Caso de Usos Visor Grafo	40
Ilustración 8: Arquitectura del Sistema	45
Ilustración 9: Diagrama de navegación	46
Ilustración 10: Cabecera Común de la Aplicación web.....	47
Ilustración 11: Pie Común de la Aplicación web.....	47
Ilustración 12: Esquema Plantilla Búsqueda	48
Ilustración 13: Esquema Plantilla Grafo	49
Ilustración 14: Árbol de ficheros de la Aplicación	53
Ilustración 15: El modelo de ciclo de Vida RAD	64
Ilustración 16: Manual - Pantalla de Búsqueda	75
Ilustración 17: Manual - Buscador.....	76
Ilustración 18: Manual - Resultado Búsqueda.....	76
Ilustración 19: Manual - Resumen Búsqueda.....	76
Ilustración 20: Manual - Navegación Búsqueda	76
Ilustración 21: Manual - Visor Grafico	77
Ilustración 22: Manual - Navegación por el Grafo I.....	77

Ilustración 23: Manual - Navegación por el Grafo II..... 78

Ilustración 24: Manual - Ficha Documento 78

Ilustración 25: Manual - Ficha Resumen Grafo 79

Índice de Tablas

Tabla 1: Requisito RF_01	35
Tabla 2: Requisito RF_02	35
Tabla 3: Requisito RF_03	35
Tabla 4: Requisito RF_04	35
Tabla 5: Requisito RF_05	35
Tabla 6: Requisito RF_06	36
Tabla 7: Requisito RF_07	36
Tabla 8: Requisito RF_08	36
Tabla 9: Requisito RF_09	36
Tabla 10: Requisito RF_10	36
Tabla 11: Requisito RF_11	36
Tabla 12: Requisito RF_12	36
Tabla 13: Requisito RF_13	36
Tabla 14: Requisito RF_14	37
Tabla 15: Requisito RF_15	37
Tabla 16: Requisito RF_16	37
Tabla 17: Requisito RF_17	37
Tabla 18: Requisito RF_18	37
Tabla 19: Requisito RF_19	37
Tabla 20: Requisito RF_20	37
Tabla 21: Requisito RF_21	37
Tabla 22: Requisito RF_22	38

Tabla 23: Requisito RNF_01	38
Tabla 24: Requisito RNF_02.....	38
Tabla 25: Requisito RNF_03.....	38
Tabla 26: Requisito RNF_04.....	38
Tabla 27: Requisito RNF_05.....	38
Tabla 28: Requisito RNF_06.....	38
Tabla 29: Ficha Caso de Uso 01	41
Tabla 30: Ficha Caso de Uso 02.....	41
Tabla 31: Ficha Caso de Uso 03.....	42
Tabla 32: Ficha Caso de Uso 04.....	42
Tabla 33: Ficha Caso de Uso 05.....	42
Tabla 34: Ficha Caso de Uso 06.....	42
Tabla 35: Ficha Caso de Uso 07	43
Tabla 36: Ficha Caso de Uso 08.....	43
Tabla 37: Ficha Caso de Uso 09.....	43
Tabla 38: Ficha Caso de Uso 10.....	43
Tabla 39: Matriz de Trazabilidad.....	44
Tabla 40: Tipo de Dato NODO	50
Tabla 41: Tipo de Dato ARISTA.....	50
Tabla 42: Tipo de Datos Grafo.....	50
Tabla 43: Prueba de Visualización.....	59
Tabla 44: Prueba PR_01.....	59
Tabla 45: Prueba PR_02.....	59
Tabla 46: Prueba PR_03.....	59

Tabla 47: Prueba PR_04.....	60
Tabla 48: Prueba PR_05.....	60
Tabla 49: Prueba PR_06.....	60
Tabla 50: Prueba PR_07.....	60
Tabla 51: Prueba PR_08.....	60
Tabla 52: Prueba PR_09.....	60
Tabla 53: Prueba PR_10.....	60
Tabla 54: Prueba PR_11.....	60
Tabla 55: Prueba PR_12.....	60
Tabla 56: Tabla de cruce prueba y requisitos.....	61
Tabla 57: Pruebas de Esfuerzo.....	62
Tabla 58: Planificación del Proyecto sobre Diagra de Gantt.....	66
Tabla 59: Resumen de personal	68
Tabla 60: Resumen de Hardware	69
Tabla 61: Resumen de Software y licencias.....	70
Tabla 62: Resumen de material fungible.....	71
Tabla 63: Resumen del presupuesto total.....	72

1. Introducción

1.1. Introducción

Las referencias a otras obras son una parte muy importante en la literatura de muchas profesiones. Todas las obras consultadas deben figurar en la sección "Referencias" o "Citas", detalladas con suficiente precisión como para que otros lectores y editores puedan consultarlas fácilmente. Ya sea una cita o una referencia los autores se apoyan en ellas para dar más crédito o base a sus textos.

En la actualidad se le da una gran importancia a la información y su forma de compartirla e actualizarla. La difusión de un texto científico es inmediata gracias a la herramienta de internet. Y el mantenimiento de la información complementaria lo es de igual manera.

Tener una lista de documentos que citan a otro es una tarea global que se actualiza de forma casi inmediata a la publicación del documento que cita o referencia a otro.

Existen en el mercado multitud de herramientas que permiten gestionar de forma global o sectorial que documentos referencian o citan a otros. Pudiéndose crear pequeñas comunidades que tengan como objetivo llevar una gestión de referencias entre documentos. También existen buscadores de lista de referencia, que permiten obtener una lista de los documentos que citan a otros.

Cuando un documento referencia o cita a otro, con gran seguridad es porque la temática está muy relacionada entre ellos. Y si uno de ellos es de interés para el lector el otro también lo puede ser. Localizar estos documentos y sobre todo los que son citados o referenciamos por muchos otros es de gran utilidad para el lector.

Cuando encadenamos documentos por su relación de referencia, partiendo de uno conocido que es referenciado por un segundo documento que a su vez lo es por un tercer documento y así hasta el nivel que deseemos. El lector puede no encontrar ese tercer o cuarto documento que tiene relación en temática con tu documento inicial y además puede que sea referenciado por muchos otros dándole una característica de relevancia.

Cuando un documento referencia a otro se crea un primer nivel de referencia y los que referencian a este segundo documento tienen un segundo nivel de relación con el primero y así sucesivamente.

Este proyecto de fin de carrera ha nacido con la intención de poder localizar de forma más rápida esos documentos, que son de primer o segundo nivel de referencia respecto a un documento ya conocido.

En la actualidad estas herramientas de gestión o búsqueda de documentos y lista documentos que citan a otro no tienen una representación gráfica, dejando la información en un ámbito contextual.

Analizadas las formas de representar estas relaciones se ha optado por una representación gráfica que ayudará al lector a localizar de forma rápida esos documentos que por ser de nivel alto de referencia no son de fácil localización. Con este proyecto se busca crear una herramienta útil para los lectores y de fácil uso.

El proyecto de fin de carrera consiste en el desarrollo de una aplicación web, a través de la cual, se represente de forma gráfica la relación de referencia entre documentos técnicos.

Para la representación gráfica se ha utilizado un grafo, ya que es la estructura relacional que se asemeja a la realidad de nuestro sistema. Con la ayuda de la representación gráfica el usuario podrá localizar documentos que al ser de nivel 2, 3 o más y que además al ser citados por muchos otros resultan relevantes y tal vez no son conocidos por el lector del documento original.

La aplicación constará de dos partes. Una de búsqueda de documentos donde se podrá realizar una consulta de un texto libre. Y una segunda donde una vez seleccionado uno de los documentos resultante de la búsqueda creará un grafo con los documentos a los que cita el documento inicial y los documentos que le citan.

En el proyecto se han utilizado para la obtención de la referencias de los documentos la web de búsqueda <http://citeseerx.ist.psu.edu/index> Web de búsqueda de documentos técnicos con información de referencias.

Esta aplicación web está desarrollada en HTML5, CSS3, JavaScript y PHP. El proyecto ha buscado que la aplicación web sea lo más fácil de implantar y se ha huido de realizar el trabajo en el lado del servidor, usando el lenguaje de programación JavaScript en el lado del cliente, tanto para la extracción de la información como la representación gráfica. Para facilitar y optimizar la programación se ha usado el Framework de JavaScript llamado JQuery y el API grafico VIS para la representación gráfica.

1.2. Objetivos

El objetivo principal de este proyecto fin de carrera es crear una herramienta que ayude a los usuarios a localizar los documentos desconocidos pero relevantes en términos de citas en un área de conocimiento.

A continuación se van a citar los objetivos a grandes rasgos de este proyecto de fin de carrera:

- Crear una aplicación web dinámica, que sea intuitiva y fácil de manejar.
- Realizar un estudio de la situación actual.
- Analizar la situación y crear una herramienta que cubra la necesidad.
- Utilizar técnicas actuales que hagan una aplicación moderna y funcional.
- Que no se requiera de herramientas externas para su uso, y así tener una fácil implantación.

1.3. Organización del documento

A lo largo del documento se va a realizar una descripción detallada de las especificaciones, capacidades y restricciones del producto. En este apartado se explica brevemente que contenido se trata en cada sección.

Capítulo 1: Se da una visión general del proyecto y los objetivos que han llevado a la realización del mismo.

Capítulo 2: Se analiza el estado del arte, se muestran las búsquedas previas para ver qué productos actuales pueden cubrir el objetivo del proyecto. Se describen las tecnologías que se van a utilizar para el desarrollo del proyecto.

Capítulo 3: Se describen las características principales del sistema. Se centra en los requisitos de software. Comenzará con una diferenciación entre requisitos funcionales y no funcionales, para ir añadiendo cada uno de ellos a su respectiva categoría. Seguido de un diseño de la aplicación a desarrollar. Para terminar en la exposición de los procesos más importantes implementados. Además de las pruebas a las que se ha sometido a la aplicación.

Capítulo 4: Se muestra el modelo del ciclo de vida, la planificación seguida en el proyecto y el presupuesto derivado de la misma.

Capítulo 5: Son las conclusiones obtenidas tras el desarrollo del sistema. También se describen posibles trabajos futuros a partir de lo realizado en el proyecto.

Anexos: Se encuentra el manual de usuario así como las referencias utilizadas y los acrónimos empleados.

2. Estado del Arte

2.1. Estudios Previos

2.1.1. Gestores de Referencias

En el mundo de las referencias bibliográficas existen una gran cantidad de aplicaciones, gratuitas y de pago, que permiten hacer una gestión de los documentos y sus referencias de forma global, permitiendo tener una lista actualizada de aquellos documentos que hacen referencia de otros.

Estas aplicaciones consiguen que la lista de los documentos que referencian a otros esta actualizada de forma rápida. Así se consigue que los lectores sepan de la existencia de documentos que pueden ser interesantes para ellos al referenciar a un documento que ya conocían o seguían.

A estas aplicaciones se las denomina gestores de referencias. Los gestores de referencias son programas que facilitan el almacenamiento, organización y control de muestras referencias bibliográficas.

Su utilización supone un gran ahorro de tiempo porque facilitan el manejo de grandes cantidades de referencias de una manera eficaz y además permiten la inserción automática de citas y la aplicación de diferentes formatos bibliográficos para generar bibliografías.

Son herramientas que permiten crear, mantener, organizar, compartir y dar forma a las referencias bibliográficas de artículos de revista, libros u otro tipo de documentos a partir de distintas fuentes de información (bases de datos, revistas, páginas web, etc.). Igualmente crearán citas y bibliografías en los documentos de trabajo con un formato normalizado (MLA, Vancouver, etc.)

Los diferentes gestores de referencias bibliográficas comparten las mismas funcionalidades básicas: almacenamiento de referencias bibliográficas; descripción, organización y recuperación de referencias; creación de bibliografías y herramientas de citación, así como compartir referencias.

Por su arquitectura se dividen en tres tipos:

- **Gestores de referencias clásicos**

Los gestores de referencias como Endnote o Reference Manager se caracterizan porque necesitan ser instalados en el ordenador.

- **Gestores de referencias web**

Nos permiten tener acceso a nuestra base de datos de referencias desde cualquier ordenador. Además, incorporan nuevas utilidades como la redifusión de contenidos, trabajos en entornos compartidos, etc. Ejemplos de este tipo son RefWorks, Endnote Web, Zotero y Mendeley.

- **Gestores de referencias sociales**

Añaden a las funcionalidades de los gestores de referencias las posibilidades que ofrecen las redes sociales para compartir y descubrir información. Ejemplos de este tipo son CiteUlike y Bibsonomy.

Se ha realizado un estudio de este tipo de aplicaciones en funciones de su relevancia y uso. Siendo MENDELEY una de las más relevantes, por sus características, así como por la cantidad de usuarios y documentos disponibles. En la actualizada multitud de Universidades Españolas están usando este gestor como apoyo a sus profesores y alumnos, fomentando el uso de la misma.

2.1.1.1 MENDELEY

Es un gestor bibliográfico que combina una versión web con una versión de Escritorio. Además incorpora funcionalidades de la Web 2.0 que permiten compartir las referencias bibliográficas con contactos y navegar por los contenidos subidos por otros usuarios. Es propietaria y gratuita. Permite gestionar y compartir documentos de investigación. Combina una aplicación de gestión de documentos PDF y de gestión de referencias con una red social online para investigadores.

Su comunidad está formada por 3 millones de usuarios y dispone de una base de datos con más de 100 millones de referencias.

Mendeley extrae automáticamente los metadatos y las referencias de los artículos desde archivos PDF. También recupera información adicional desde Crossref, PubMed, ArXiv, etc.

Mendeley ha ganado varios premios: Plugg.eu "European Start-up of the Year 2009",⁵ 6 TechCrunch Europas "Best Social Innovation Which Benefits Society 2009",⁷ y The Guardian la situó en el puesto número 6 en el "Top 100 tech media companies" del Reino Unido.⁸

Características

- Escritorio Mendeley, basado en Qt, se puede ejecutar tanto en Windows, como en Mac y Linux.
- Extracción automática de metadatos de documentos PDF.
- Copias de seguridad y sincronización entre varios equipos y con una cuenta en línea privada.

- Visor de documentos PDF con notas adhesivas, selección de texto y lectura a pantalla completa.
- Búsqueda completa de texto a través de documentos.
- Filtrado inteligente, etiquetado y cambio de nombre de archivos PDF.
- Citas y bibliografías en Microsoft Word, OpenOffice y LibreOffice
- Importación de documentos y trabajos de investigación de sitios web externos (por ejemplo, PubMed, Google Scholar, Arxiv, etc.) haciendo uso del bookmarklet del navegador.
- Mendeley web soporta COinS y el bookmarklet del navegador importa de cualquier sitio que soporte COinS.
- Compartir y colaborar en grupo, anotaciones en los documentos.
- Características de redes sociales (seguimiento de investigadores con ideas afines, noticias).
- Estadísticas sobre los documentos, autores y publicaciones más leídos.

2.1.1.2 Otros Gestores de Referencia

EndNote

El gestor de referencias bibliográficas EndNote Web se encuentra disponible para todos los usuarios de la UCA¹ dentro de ISI Web of Knowledge de forma gratuita gracias a la licencia nacional de la FECYT².

Este programa permite crear una base de datos personalizada de hasta 10.000 referencias y organizar la bibliografía personal. Además, ofrece cientos de formatos diferentes para las referencias bibliográficas e importar desde una enorme cantidad de fuentes y bases de datos, así como compartir referencias con otros usuarios de Endnote. La funcionalidad de %Cite-While-You-Write+ permite rápidamente importar y formatear la referencia en un documento de texto.

Zotero

Gestor de referencias que funciona en entornos web. Es de software libre, con licencia abierta, se trata de una extensión de Mozilla y FireFox, y sólo funciona con este navegador. Es software libre y está disponible en más de 30 idiomas.

El gestor permite obtener las referencias, administrar, citar y compartir trabajos de investigación de cualquier origen y procedencia desde el propio navegador. Además, permite exportar los datos en diferentes estilos de citas, informes y bibliografías.

¹ UCA Universidad de Cádiz

² FECYT Fundación Española para la Ciencia y la Tecnología

Refworks³

La Biblioteca de la UCA, a través de suscripción, ofrece este gestor a todos sus usuarios. La aplicación permite crear una base de datos en línea con referencias extraídas automáticamente desde muchas bases de datos y portales. La información se puede organizar en carpetas que se pueden compartir, generar bibliografías, escribir nuestro trabajo e insertar las citas automáticamente, etc.

Citeulike

Es un gestor de referencias bibliográficas que permite almacenar, organizar y compartir los artículos que esté leyendo. Es un servicio web donde la colección de referencias está guardada en el servidor de Citeulike, de forma que garantiza el acceso desde cualquier ordenador con conexión a Internet.

Es ~~%~~ social+ porque deja compartir la biblioteca de referencias y saber quién está leyendo el mismo artículo, permitiendo descubrir literatura relevante en nuestro campo de investigación que desconocíamos.

2.1.2. Buscadores de Documentos Académicos

En el mundo de los buscadores de documentos académicos existen una gran cantidad de aplicaciones, gratuitas y de pago, que permiten buscar documentos de índole científico. Además se podrá conseguir información útil como la cantidad de veces que es referencia o a que otros documentos referencia y otros documentos de temática relacionada. A continuación vamos a describir los buscadores de documentos académicos más usados.

2.1.2.1 CiteSeerX

CiteSeerX es un motor de búsqueda público y biblioteca digital para publicaciones académicas y científicas con especial énfasis en ciencias de la computación. La idea detrás de CiteSeerX es principalmente la misma que su antecesor CiteSeer; sin embargo, ha sido construido usando una nueva infraestructura basada en código abierto, SeerSuite, así como nuevos algoritmos con sus respectivas implementaciones. CiteSeerX ha sido desarrollado, en el Colegio de Ciencias de la Información y Tecnología de la Universidad Estatal de Pensilvania, por los investigadores Dr. Isaac Council y Dr. Lee Giles.

³ Refworks dejará de estar operativo el 31 de Diciembre de 2015.

Además de los objetivos propuestos por su predecesor, CiteSeer, búsqueda y captura activa de documentos académicos y científicos en WEB para ser indexados usando el método autónomo de análisis de citas; y así permitir las búsquedas por cita o por la clasificación de los documentos basado en dicho análisis, CiteSeerX intenta proveer recursos tales como algoritmos, datos, metadatos, servicios, técnicas y software que puede ser utilizado para promover el uso y desarrollo de nuevas bibliotecas digitales.

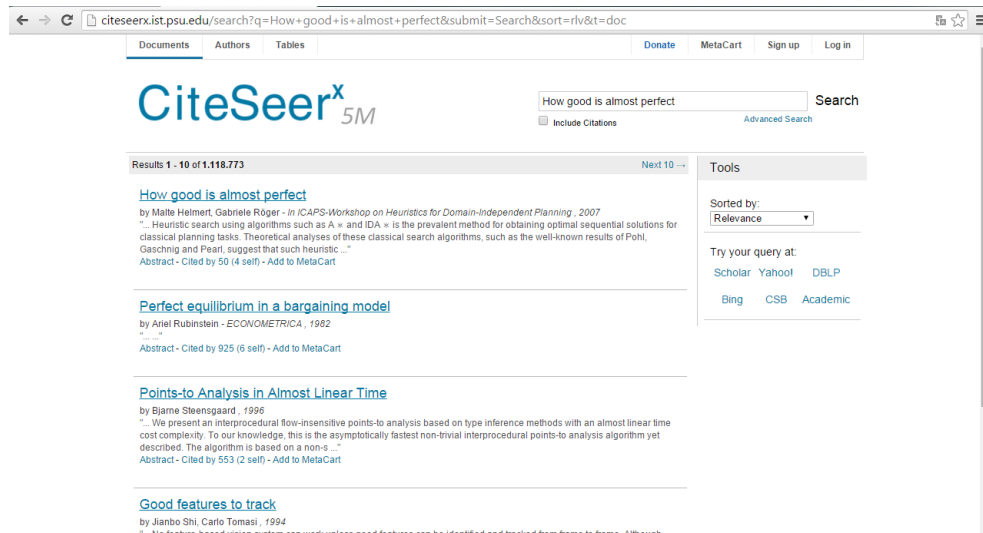


Ilustración 1: Resultado Búsqueda en CiteSeerX

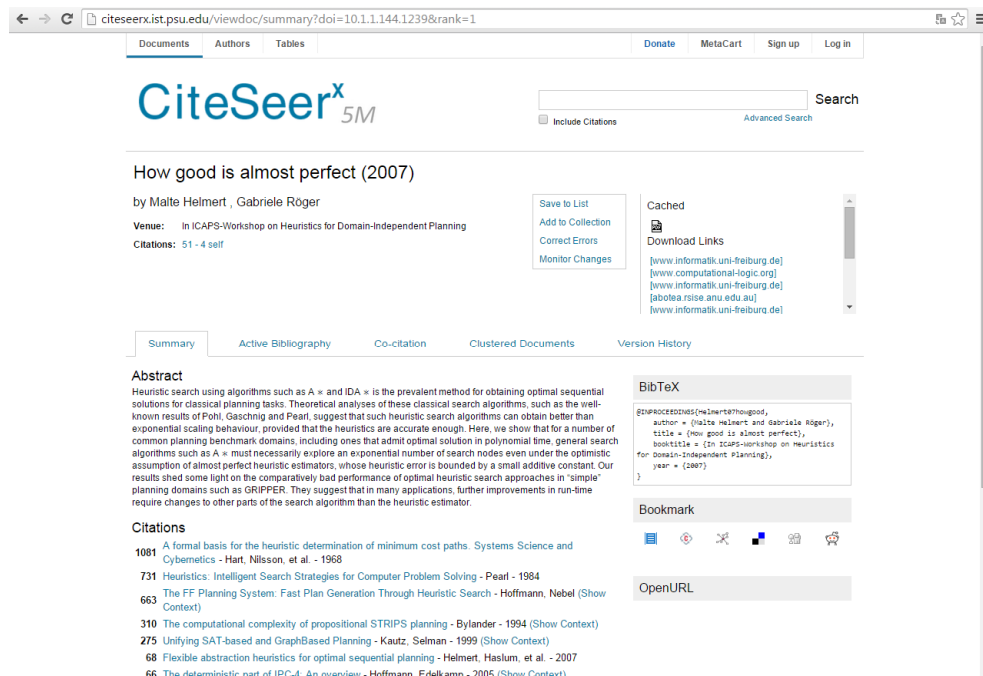


Ilustración 2: Ficha de Documento en CiteSeerX

2.1.2.2 Otros buscadores de Documentos Académicos

Scirus

Scirus era un motor de búsqueda específico de contenido científico. Fue el primer buscador de su tipo, creado por la editorial holandesa Elsevier. Semejante a CiteSeerX y Google Scholar, estaba enfocado a la búsqueda de información científica en bases de datos especializadas. A diferencia de CiteSeer, Scirus no se centraba sólo en ciencias informáticas, y tampoco todo su contenido era de libre acceso. Algunos resultados se podían encontrar también en PubMed, o en cualquier revista de Elsevier, lo que requería de una suscripción para poder acceder al mismo.

En octubre de 2013, Elsevier anunció que cerraría Scirus a comienzos de 2014. En febrero de 2014 se cerró definitivamente el servicio, lo que ha supuesto la desaparición de Scirus.

Google Scholar

Google Académico es un buscador de Google especializado en artículos de revistas científicas, enfocado en el mundo académico, y soportado por una base de datos disponible libremente en Internet que almacena un amplio conjunto de trabajos de investigación científica de distintas disciplinas y en distintos formatos de publicación.

Fue lanzado al público en versión Beta el 18 de noviembre de 2004. El índice GS incluye las revistas más leídas en el mundo científico con excepción de Elsevier. Google Académico es similar en función a los ya disponibles Scirus de Elsevier y CiteSeer.

Jerarquiza los resultados usando un algoritmo similar al que utiliza Google para las búsquedas generales, aunque también usa como señal de "calidad" la revista en la que se ha publicado. Los resultados incluyen libros técnicos, así como un enlace a otros artículos que citan el artículo señalado. Esto es una herramienta interesante para la investigación, ya que permite encontrar nueva información (más actualizada) a partir de un artículo conocido.

Microsoft Academic Search (MAS)

Es una herramienta para explorar las conexiones en la investigación, a través de los registros de publicación de los investigadores, universidades y organizaciones de investigación. Este motor de búsqueda, en versión beta, se lanzó hace algo más de un año, centrado en el ámbito de las ciencias de la computación. Actualmente, recoge 15 áreas, incluyendo Ciencias Sociales, Arte y Humanidades, Ciencias del Medio

Ambiente y Economía y Empresa. Cada dominio tiene subdominios, que permiten búsquedas más concretas. Cubre más de 38 millones de publicaciones y más de 19 millones de autores, con actualizaciones semanales.

Además, de la búsqueda sencilla, en la búsqueda avanzada, ofrece la posibilidad de recuperar por Autor, Conferencia, Revista, Organización, Año y DOI⁴. Una vez obtenido el primer nivel de resultados, se puede acotar por especialidad y una fecha concreta de publicación o un intervalo de fechas.

Desde la página inicial de búsquedas, ofrece también la posibilidad de obtener los datos desde diferentes perspectivas:

- Navegar por comunidades de organizaciones y autores, dentro de una misma área temática.
- Comparar dos organizaciones; ofrece además recuentos de citas, palabras clave, autores principales y más.
- Muestra las relaciones de un autor con su comunidad científica.
- Búsqueda de conferencias por área temática, fecha y hora. Permite localizar convocatorias científicas a seis meses vista (conferencias, encuentros, seminarios, etcétera) que pueden filtrarse por especialidad y/o por región geográfica.
- Muestra las redes de colaboración de un autor concreto. Las coautorías.
- Descubre qué publicaciones han citado una publicación concreta.
- Permite visualizar las tendencias de investigación a través de un gráfico de áreas apiladas interactivo.
- Muestra las conexiones de dos investigadores, a través de sus co-autores.

Dispone de un servicio de seguimiento en redes sociales como Twitter sobre el propio buscador y se pueden compartir los resultados de nuestras búsquedas en Facebook o Twitter.

⁴ <http://www.doi.org/>

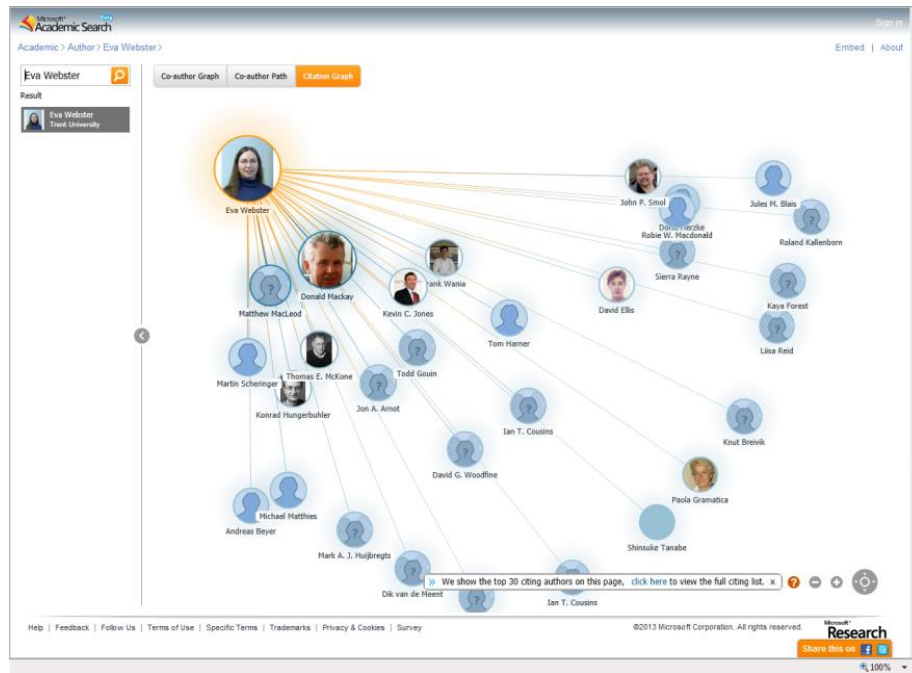


Ilustración 3: Ejemplo Grafico de MAS

Además, si te registras, accedes a otros servicios de valor añadido y contribuyes al proyecto.

Introduce un visor gráfico de referencias por autor, cosa que los anteriores buscadores no tienen.

2.1.3 Grafo Social

El grafo social es un término acuñado por Mark Zuckerberg, fundador de Facebook⁵, que originalmente se refería a la red social de relaciones entre los usuarios del servicio de redes sociales proporcionados por Facebook. Este término se ha descrito como «el mapa global de todo el mundo y cómo están relacionados». El término fue acuñado en la conferencia Facebook f8 el 24 de mayo de 2007, cuando fue utilizado para explicar que la plataforma de Facebook se beneficiaría del gráfico social, tomando ventaja de las relaciones entre los individuos que Facebook ofrece para ofrecer una rica experiencia en línea. Ha sido la definición más amplia para referirse a un gráfico social de todos los usuarios de Internet.

El grafo social es un concepto abstracto que describe las relaciones entre las personas en línea, en contraposición a la idea de una red social, que describe las relaciones en el mundo real. Los dos conceptos son muy similares, pero existen algunas diferencias de menor importancia. Por ejemplo, el gráfico social es digital y, más importante aún, se define explícitamente por todas las conexiones en cuestión.

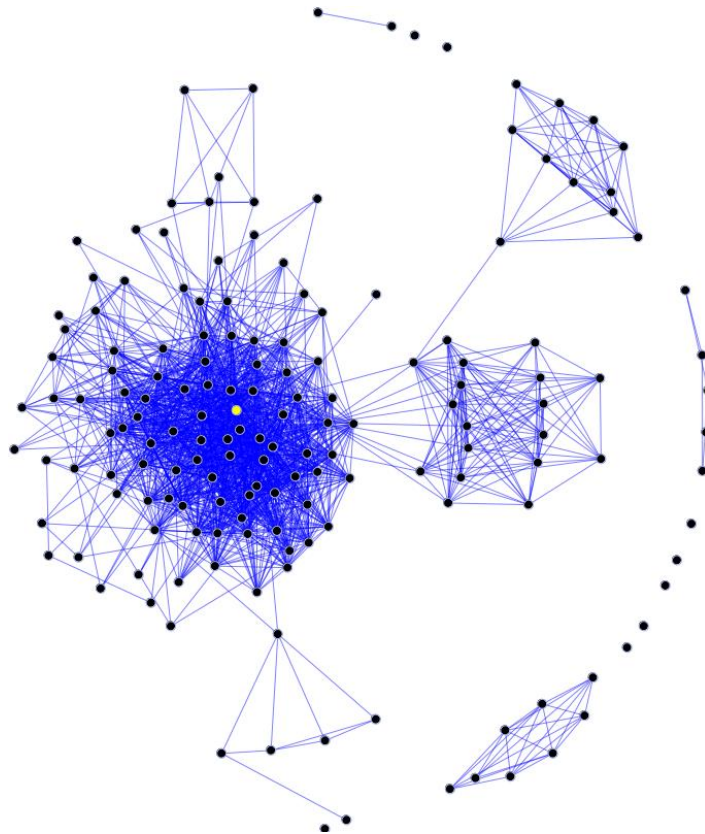


Ilustración 4: Ejemplo de un Grafo Social

⁵ <https://www.facebook.com/>

2.2. Tecnologías Aplicadas

2.2.1. HTML

HTML son las siglas de HyperText Markup Language (Lenguaje de Marcas de Hipertexto), es el utilizado para la construcción de páginas web. Su uso está destinado a describir la estructura y contenido de los sitios web en forma de texto, así como para complementar el texto con objetos tales como imágenes. Este lenguaje se construye mediante etiquetas que delimitan las secciones de la web y alojan la información. Normalmente suele estar implementado junto con hojas de estilo llamadas CSS que ayudan a definir los aspectos visuales de la web. HTML puede incluir scripts (JavaScript, PHP) que afectan al comportamiento de navegadores web y otros procesadores de HTML.

HTML5

HTML5 establece una serie de nuevos elementos y atributos que reflejan el uso típico de los sitios web modernos. Algunos de ellos son técnicamente similares a las etiquetas `<div>` y ``, pero tienen un significado semántico, como por ejemplo `<nav>` (bloque de navegación del sitio web) y `<footer>`. Otros elementos proporcionan nuevas funcionalidades a través de una interfaz estandarizada, como los elementos `<audio>` y `<video>`. Mejora el elemento `<canvas>`, capaz de renderizar elementos 3D en los navegadores más importantes (Firefox, Chrome, Opera, Safari e Internet Explorer). También hay un renovado énfasis en la importancia del scripting DOM para el comportamiento de la web 2.0.⁶

Ya que se va a desarrollar una aplicación web es imprescindible el uso de HTML. Por sus nuevos interfaces gráficos se ha optado por el uso de HTML5 ante otras versiones anteriores de HTML.

2.2.2. Hoja de estilos CSS

CSS o lo que es lo mismo, hojas de estilo en cascada, es un lenguaje utilizado para definir la presentación de un documento escrito en HTML o XML. El W3C es el encargado de formular las especificaciones de las hojas de estilo que servirán de estándar para los navegadores. La idea que se mantiene para el desarrollo de las hojas de estilo es separar la estructura del documento de su presentación, lo que hace más fácil la modificación de los estilos en caso de ser necesario. Cuando se utiliza CSS, las etiquetas no deben proporcionar información acerca de la visualización del documento, sino marcar la estructura. La información del estilo, separada en una hoja de estilo, especifica cómo se ha de mostrar una etiqueta; color, fuente, alineación del texto o tamaño.

⁶ El término Web 2.0 comprende aquellos sitios web que facilitan el compartir información, la interoperabilidad, el diseño centrado en el usuario y la colaboración en la WWW.

CSS3

La especificación de CSS3 tiene interesantes novedades respecto de su versión anterior que permitirán hacer webs más elaboradas y más dinámicas, con mayor separación entre estilos y contenidos. Dando soporte a muchas necesidades de las webs actuales, sin tener que recurrir a trucos de diseñadores o lenguajes de programación.

Para tener que evitar usar trucos o añadir más programación a nuestra aplicación se ha optado por usar esta versión de capa de estilo.

2.2.3. PHP

El lenguaje PHP es un lenguaje de programación diseñado y orientado a la creación de páginas dinámicas. Es utilizado principalmente en interpretación del lado del servidor pero actualmente puede ser utilizado desde una interfaz de línea de comandos o en la creación de otro tipo de aplicaciones en las que se encuentran aquellas con interfaz gráfica. La mayor fuerza de PHP reside en que es un lenguaje preparado para soportar accesos a muchos tipos de bases de datos como Oracle, ODBC, DB2, SQLServer...etc.

PHP nos permite obtener código HTML de una URL aunque esta no esté en nuestro dominio. Esta característica ha hecho que sea necesario su uso en el proyecto.

2.2.4. JavaScript

JavaScript es un lenguaje basado en objetos y guiado por eventos, utilizado para acceder a objetos en aplicaciones. Se utiliza integrado en un navegador web permitiendo el desarrollo de interfaces de usuario mejoradas y páginas web dinámicas. Todos los navegadores actuales interpretan el código de JavaScript integrado dentro de las páginas web. Entre las acciones típicas que se pueden realizar en JavaScript tenemos dos vertientes. Por un lado los efectos especiales sobre páginas web, para crear contenidos dinámicos y elementos de la página que tengan movimiento, cambien color o cualquier otro dinamismo. Por el otro, JavaScript nos permite ejecutar instrucciones como respuesta a las acciones del usuario, con lo que podemos crear páginas interactivas con programas como calculadoras, agendas o tablas de cálculo.

JavaScript nos permite acceder al DOM⁷ de la página web y modificarlo sin necesidad de llamar al servidor. Esto da al usuario un aspecto de aplicación clásica de escritorio. Estas características son las que hacen que sea utilizado en nuestro proyecto, dando un aspecto dinámico.

⁷ 'Modelo de Objetos del Documento' - una interfaz de programación de aplicaciones (API)

2.2.5. JQuery

jQuery es un framework de JavaScript que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX a páginas web. JQuery es la biblioteca de JavaScript más utilizada.

jQuery es software libre y de código abierto, permitiendo su uso en proyectos libres y privados. jQuery, al igual que otras bibliotecas, ofrece una serie de funcionalidades basadas en JavaScript que de otra manera requerirían de mucho más código, es decir, con las funciones propias de esta biblioteca se logran grandes resultados en menos tiempo y espacio. Es un lenguaje sencillo, si lo comparamos con sus homólogos y dispone de un amplio conjunto de funciones para animar el contenido de la página de forma muy sencilla. La intención es ahorrar código a los programadores creando funciones sencillas que realizan varias tareas.

Por defecto integra funcionalidades para trabajar con AJAX, lo cual hace posible que la página sea más dinámica. Tiene la ventaja de que el servidor lo enviará al cliente la primera vez que visite una página del sitio. En siguientes páginas el cliente ya tendrá el archivo de jQuery, por lo que no necesitará transferirlo y lo tomará de la caché. Con lo que la carga de la página sólo se verá afectada por el peso de este framework de jQuery una vez por usuario.

Para facilitar nuestra labor en la programación en JavaScript se va a utilizar este framework en vez de otros similares ya que sus características e implantación en otras web son mayores. Además porque va a simplificar en gran medida tareas en el desarrollo.

2.2.6. Librería Gráfica VISJP

Existen librerías JavaScript que hacen magia en nuestros proyectos y nos facilitan la vida. Vis.js es una librería de visualización para proyectos Web que ha sido diseñada para que sea fácil de utilizar y gestionar grandes cantidades de datos dinámicamente, además de posibilitar la interacción del usuario con dichos datos.

Vis.js nos permite realizar distintos tipos de gráficos:

- **Redes:** conexiones y relaciones entre diferentes nodos.
- **Líneas temporales:** ver qué cambios ha habido a lo largo del tiempo en un determinado contexto.
- **Gráficas 2D:** gráficas en dos dimensiones, con ejes x y y .
- **Gráficas 3D:** gráficas tridimensionales para representar 3 coordenadas.

Es muy configurable y dinámica, además de que el propio usuario puede configurarla a su gusto. Vis.js se ejecuta en FIREFOX, Chrome, Opera, Safari, Internet Explorer 9+ y

otros navegadores en dispositivos móviles, ya que la librería tiene funciones táctiles para teléfonos, tablets, etc.

2.2.7. Ajax

Ajax es el acrónimo de Asynchronous JavaScript And XML. Es una técnica de desarrollo web para crear aplicaciones interactivas. El concepto es cargar y renderizar una página, luego mantenerse en esa página mientras scripts y rutinas van al servidor buscando, en background, los datos que son usados para actualizar los gráficos de la página.

Es una tecnología asíncrona, los datos se obtienen de la base de datos y se cargan en segundo plano sin interferir en el uso de la aplicación para el usuario. En este tipo de aplicaciones se introduce un intermediario, un motor Ajax, entre el usuario y el servidor. En vez de cargar una página web, al inicio de la sesión, el navegador carga al motor Ajax para que sea el responsable de renderizar la interfaz y comunicarse con el servidor en nombre del usuario. El motor Ajax permite que la interacción del usuario con la aplicación suceda asíncronamente, de esta forma el usuario nunca estará mirando una ventana en blanco del navegador.

La única manera de realizar el proyecto sin Ajax sería mediante la creación de un formulario, cuando el usuario envíe los datos del formulario es cuando se realizaría la conexión con la base de datos y se mostraría por pantalla la página que devuelve el servidor. Con este método se recargaría la página bien saltando a una nueva o a la misma, el usuario tendría que soportar la espera de cada recarga y este proceso podría resultar lento puesto que la información HTML se descarga por duplicado.

Por todos estos motivos hemos utilizado Ajax y por tanto un código JavaScript es el utilizado para crear un objeto XMLHttpRequest al enviar el formulario, los datos se envían al servidor pero no se recarga la página, posteriormente el servidor responde la petición y una función JavaScript es la que valora la respuesta del servidor, si la respuesta es válida se imprime el texto al usuario. Este método es bastante más rápido porque el código HTML de la página de confirmación del formulario ya no se tiene que descargar.

A continuación vamos a citar las diez razones por la que considerar el uso de Ajax que vienen comentadas en numerosos sitios web dedicados a desarrolladores

1. Basado en los estándares abiertos: está formado por las tecnologías más usadas y que soportan los navegadores más utilizados de internet.
2. Usabilidad: Elimina tener que refrescar el navegador cuando se realiza una petición de datos al servidor.
3. Válido en cualquier plataforma y navegador: Los navegadores basados en Mozilla y FireFox son los más sencillos para programar aplicaciones web Ajax

pero ahora es posible que esas aplicaciones funcionen en los navegadores más modernos.

4. Beneficia las aplicaciones Web: Cada vez se utilizan más las aplicaciones web puesto que requieren un menor coste de creación y facilitan el soporte y mantenimiento. Ajax ayuda a estas aplicaciones a mejorar y conseguir un mejor resultado para el usuario.
5. No es difícil su utilización: Ajax está basada en los estándares que se utilizan y por tanto los desarrolladores no requieren de un gran esfuerzo para su aprendizaje.
6. Compatible con Flash: Existen ventajas y desventajas para ambas tecnologías por lo que el uso combinado ofrece buenas alternativas.
7. Adoptado por los "gordos" de la tecnología web: El mercado acepta y valida el uso de esta tecnología. Grandes de la industria como Google, Yahoo, Amazon, Microsoft la utilizan.
8. Web 2.0: Una de las claves de la web 2.0 es usar la red como plataforma de desarrollo de aplicaciones siendo muy importante la iteración de los usuarios.
9. Es independiente del tipo de tecnología de servidor que se utilice: Ajax es compatible con cualquier tipo de servidor estándar y lenguaje de programación web.
10. Mejora la estética de la web: Con Ajax se pueden realizar aplicaciones que podrían ser de escritorio ya que permite combinar la imaginación de los desarrolladores con la usabilidad.

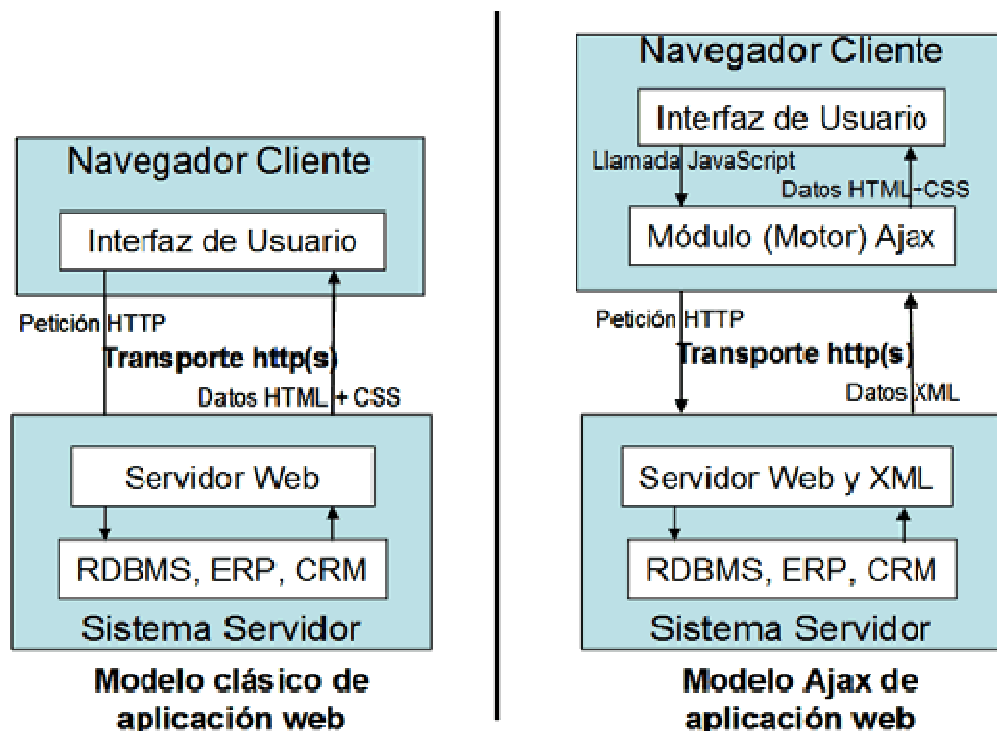


Ilustración 5: Comparativa Ajax-Tradicional

La figura anterior muestra la comparativa entre el modelo tradicional para el desarrollo de aplicaciones web y el modelo utilizado para Ajax. Se puede observar que la gran diferencia reside en el motor Ajax que actúa de intermediario entre el cliente y el servidor.

3 Análisis, Diseño e Implementación

3.1 Análisis

3.1.1 Requisitos Usuarios

Desde la adjudicación del proyecto se han recogido notas de las conversaciones mantenidas con el profesor Tomas, con el fin de definir las necesidades de la herramienta de software a fabricar. Requisitos que a continuación se enumeran, han sido recogidos en lenguaje natural. Hemos separado el tipo de requisitos en dos grandes grupos: funcionales y no funcionales.

Requisitos no funcionales

- La arquitectura será cliente-servidor.
- El servidor deberá tener conexión a internet.
- El servidor deberá interpretar el lenguaje PHP.
- La herramienta será ejecutable para computadoras cliente con cualquier sistema operativo: Microsoft Windows, Linux, Macintosh, etc.
- La herramienta será funcional para las últimas versiones de navegadores. Para el Internet Explorer se requerirá de versión 9 o superior.
- El navegador deberá tener habilitado la ejecución de código JavaScript.
- No será necesaria la instalación de ningún archivo complementario a la propia aplicación (controladores, entornos de desarrollo, etc.) que no sea un navegador para la ejecución de la misma.
- La herramienta será totalmente autónoma e independiente de cualquier otro sistema de información.
- El número de usuarios que soliciten la herramienta al servidor estará limitado por las capacidades de dicho servicio, esto es, el ancho de banda, la memoria, capacidad de proceso y capacidad de gestión de usuarios por parte de la máquina servidora.
- Si el usuario ha guardado la página web de la herramienta en su propia máquina, la podrá utilizar mediante el navegador sin tener que volver a conectar con el servidor. Por tanto, no se ejecutará código alguno en el servidor, excepto o las propias peticiones HTTP solicitantes para que el cliente acceda a la herramienta bien para su ejecución directa, bien para guardarla.
- La herramienta será eminentemente gráfica.
- La información se obtendrá de la web: <http://citeseerx.ist.psu.edu/index>.
- No se guardará ningún dato de los usuarios, en referencia a la Ley de Protección de datos, por tanto no son necesarios requerimientos de seguridad por parte de la herramienta, exceptuando efectuar una copia de seguridad de la propia herramienta.

Requisitos funcionales

- El usuario realizará una consulta de texto libre igual a la consulta de la web <http://citeseerx.ist.psu.edu/index>.
- El resultado de la búsqueda se paginará en bloques de 10 documentos.
- Se mostrará el número de documentos coincidente con la búsqueda, así como el número del bloque de la página mostrada.
- Se podrá navegar por las páginas de resultados con página siguiente y página atrás.
- La plantilla de búsqueda no se cerrará al seleccionar un documento para ver su grafo.
- La representación gráfica de los datos recogidos será un grafo con un nodo por documento.
- Una vez seleccionado un documento se generará un grafo con la información extraída.
- Sólo se extraerá, inicialmente, un nivel a partir del documento seleccionado. Sólo los documentos a los que hace referencia y los que le referencian.
- Cada nodo del grafo tendrá un tamaño en función del número de documentos que le referencian.
- Para evitar grandes diferencias el tamaño de cada nodo será en una escala de 1 a 10.
- La escala será proporcional al valor máximo y mínimo de las referencias de los nodos.
- Cada nodo tendrá un color que indicará si es el documento seleccionado, si lo referencia este o si es referenciado por él.
- Cada nodo tendrá un tooltip con el nombre del documento y el número de documento que le referencian.
- Al seleccionar un nodo se mostrará una ficha con los datos siguientes datos, siempre que se puedan obtener:
 - Nombre
 - Resumen
 - Año de creación
 - Autores
- Desde la ficha de un nodo se podrá realizar las siguientes tareas:
 - Generar un nuevo grafo del documento de la ficha abierta.
 - Ver la ficha del documento en <http://citeseerx.ist.psu.edu/index>
- Se mostrará un resumen de los nodos y vértices del grafo creado en el grafo.
- Sobre el grafo se deberá permitir:
 - Hacer zoom + y -.
 - Mover los nodos de posición.
 - Mover el grafo.
- El grafo deberá ser fluido para al menos 500 nodos y 1000 vértices. A partir de esa cantidad no se podrá asegurar un uso rápido y funcional.

A continuación se especificarán los requisitos de usuario dividiéndolos en dos categorías, requisitos funcionales y requisitos no funcionales. Estos requisitos se

especifican a continuación siguiendo un formato tabular en el cual se incluye el siguiente contenido para cada uno de ellos:

ID: indica de manera unívoca a un requisito. Nomenclatura: RF_XX para los requisitos funcionales y RNF _XX para los no funcionales; donde XX corresponde al número de requisito.

DESCRIPCIÓN: especificación del requisito.

NECESIDAD: relevancia del requisito para el proyecto. Nomenclatura: valores entre 1 y 5 siendo 5 la necesidad más alta y 1 la más baja.

PRIORIDAD: establece la prioridad del requisito dentro del proyecto. Nomenclatura: valores entre 1 y 5 siendo 5 la prioridad más alta y 1 la más baja.

ESTABILIDAD: indica la sensibilidad del requisito a ser modificado. Nomenclatura: %Estable+y %No Estable+

ID	RF_01
DESCRIPCIÓN	La obtención de la información se realizará a la web: CiteSeerX
NECESIDADES	5
PRIORIDAD	5
ESTABILIDAD	Estable

Tabla 1: Requisito RF_01

ID	RF_02
DESCRIPCIÓN	El usuario realizará consultas de texto libre al estilo de la web CiteSeerX
NECESIDADES	5
PRIORIDAD	5
ESTABILIDAD	Estable

Tabla 2: Requisito RF_02

ID	RF_03
DESCRIPCIÓN	El usuario tras la búsqueda seleccionará un documento para generar el grafo.
NECESIDADES	5
PRIORIDAD	5
ESTABILIDAD	Estable

Tabla 3: Requisito RF_03

ID	RF_04
DESCRIPCIÓN	Se podrá paginar por la búsqueda con next y prev.
NECESIDADES	5
PRIORIDAD	5
ESTABILIDAD	Estable

Tabla 4: Requisito RF_04

ID	RF_05
DESCRIPCIÓN	El resultado de la búsqueda se mostrará en bloques de 10 documentos.
NECESIDADES	3
PRIORIDAD	5
ESTABILIDAD	Estable

Tabla 5: Requisito RF_05

ID	RF_06
DESCRIPCIÓN	Se mostrará el número total de documentos encontrados así como el bloque de documentos actual.
NECESIDADES	3
PRIORIDAD	3
ESTABILIDAD	Estable

Tabla 6: Requisito RF_06

ID	RF_07
DESCRIPCIÓN	La ventana con el resultado de la búsqueda no se cerrará tras seleccionar un documento.
NECESIDADES	4
PRIORIDAD	4
ESTABILIDAD	Estable

Tabla 7: Requisito RF_07

ID	RF_08
DESCRIPCIÓN	Los datos necesarios para el grafo se extraerán de la web CiteSeerX
NECESIDADES	5
PRIORIDAD	5
ESTABILIDAD	Estable

Tabla 8: Requisito RF_08

ID	RF_09
DESCRIPCIÓN	Inicialmente se obtendrá un NIVEL de referencias a partir del documento seleccionado.
NECESIDADES	3
PRIORIDAD	3
ESTABILIDAD	Estable

Tabla 9: Requisito RF_09

ID	RF_10
DESCRIPCIÓN	Cada nodo representará un documento.
NECESIDADES	5
PRIORIDAD	5
ESTABILIDAD	Estable

Tabla 10: Requisito RF_10

ID	RF_11
DESCRIPCIÓN	Se creará una escala de 1 a 10 para el tamaño de los nodos.
NECESIDADES	5
PRIORIDAD	5
ESTABILIDAD	Estable

Tabla 11: Requisito RF_11

ID	RF_12
DESCRIPCIÓN	Cada nodo tendrá un color en función si es el documento seleccionado, un documento referenciado o uno que le referencia.
NECESIDADES	5
PRIORIDAD	5
ESTABILIDAD	Estable

Tabla 12: Requisito RF_12

ID	RF_13
DESCRIPCIÓN	Cada nodo del grafo tendrá un tamaño en función del número de documento que lo referencien (número de citas).
NECESIDADES	5
PRIORIDAD	5
ESTABILIDAD	Estable

Tabla 13: Requisito RF_13

ID	RF_14
DESCRIPCIÓN	Cada nodo tendrá un tooltip con el nombre del documento y el número de documento que lo referencia.
NECESIDADES	5
PRIORIDAD	5
ESTABILIDAD	Estable

Tabla 14: Requisito RF_14

ID	RF_15
DESCRIPCIÓN	Al seleccionar un nodo se mostrará una ficha con los siguientes datos: nombre, resumen, autor y año.
NECESIDADES	5
PRIORIDAD	5
ESTABILIDAD	Estable

Tabla 15: Requisito RF_15

ID	RF_16
DESCRIPCIÓN	Desde la ficha del documento se podrá generar su grafo.
NECESIDADES	5
PRIORIDAD	5
ESTABILIDAD	Estable

Tabla 16: Requisito RF_16

ID	RF_17
DESCRIPCIÓN	Desde la ficha del documento se podrá ver la ficha del documento en la web CiteSeerX
NECESIDADES	5
PRIORIDAD	5
ESTABILIDAD	Estable

Tabla 17: Requisito RF_17

ID	RF_18
DESCRIPCIÓN	Se mostrará un resumen del número de nodos y vértices del grafo.
NECESIDADES	5
PRIORIDAD	5
ESTABILIDAD	Estable

Tabla 18: Requisito RF_18

ID	RF_19
DESCRIPCIÓN	Sobre el grafo se podrá hacer Zoom +
NECESIDADES	5
PRIORIDAD	5
ESTABILIDAD	Estable

Tabla 19: Requisito RF_19

ID	RF_20
DESCRIPCIÓN	Se podrá mover todo el grafo.
NECESIDADES	5
PRIORIDAD	5
ESTABILIDAD	Estable

Tabla 20: Requisito RF_20

ID	RF_21
DESCRIPCIÓN	Sobre el grafo se podrá mover un nodo de la posición inicial.
NECESIDADES	5
PRIORIDAD	5
ESTABILIDAD	Estable

Tabla 21: Requisito RF_21

ID	RF_22
DESCRIPCIÓN	Se podrá paginar por la búsqueda con siguiente.
NECESIDADES	5
PRIORIDAD	5
ESTABILIDAD	Estable

Tabla 22: Requisito RF_22

ID	RNF_01
DESCRIPCIÓN	La aplicación se programará en HTML5, PHP y JavaScript con CSS3 para el diseño.
NECESIDADES	5
PRIORIDAD	5
ESTABILIDAD	Estable

Tabla 23: Requisito RNF_01

ID	RNF_02
DESCRIPCIÓN	El navegador de ejecución será IE9, Mozilla 10 y Chrome 18. Mostrando un mensaje si no cumple esta condición
NECESIDADES	5
PRIORIDAD	5
ESTABILIDAD	Estable

Tabla 24: Requisito RNF_02

ID	RNF_03
DESCRIPCIÓN	El idioma de la aplicación será el castellano.
NECESIDADES	5
PRIORIDAD	5
ESTABILIDAD	Estable

Tabla 25: Requisito RNF_03

ID	RNF_04
DESCRIPCIÓN	No se guardará ningún dato del usuario para cumplir la ley de protección de datos.
NECESIDADES	5
PRIORIDAD	5
ESTABILIDAD	Estable

Tabla 26: Requisito RNF_04

ID	RNF_05
DESCRIPCIÓN	El navegador tendrá que tener habilitado la ejecución de JavaScript.
NECESIDADES	5
PRIORIDAD	5
ESTABILIDAD	Estable

Tabla 27: Requisito RNF_05

ID	RNF_06
DESCRIPCIÓN	No será necesario instalar ningún complemento adicional que no sea el navegador.
NECESIDADES	5
PRIORIDAD	5
ESTABILIDAD	Estable

Tabla 28: Requisito RNF_06

ID	RNF_07
DESCRIPCIÓN	El límite de usuarios lo fijará el servidor donde este publicada la aplicación.
NECESIDADES	5
PRIORIDAD	5
ESTABILIDAD	Estable

Tabla 29: Requisito RNF_07

3.1.2 Casos de Uso

Un caso de uso es una técnica usada para realizar la captura de requisitos de un sistema informático. Cada caso de uso muestra los escenarios que indican cómo se interactúa entre el sistema y los usuarios del mismo. En este caso tenemos un único tipo de usuario diferenciado:

- Usuario Aplicación WEB: Aquellas personas que pertenecen a la organización en la que se utilice esta aplicación.

3.1.2.1 Descripción gráfica

En este apartado se muestran los diagramas de casos de uso representativos de las principales funcionalidades del usuario de la aplicación.

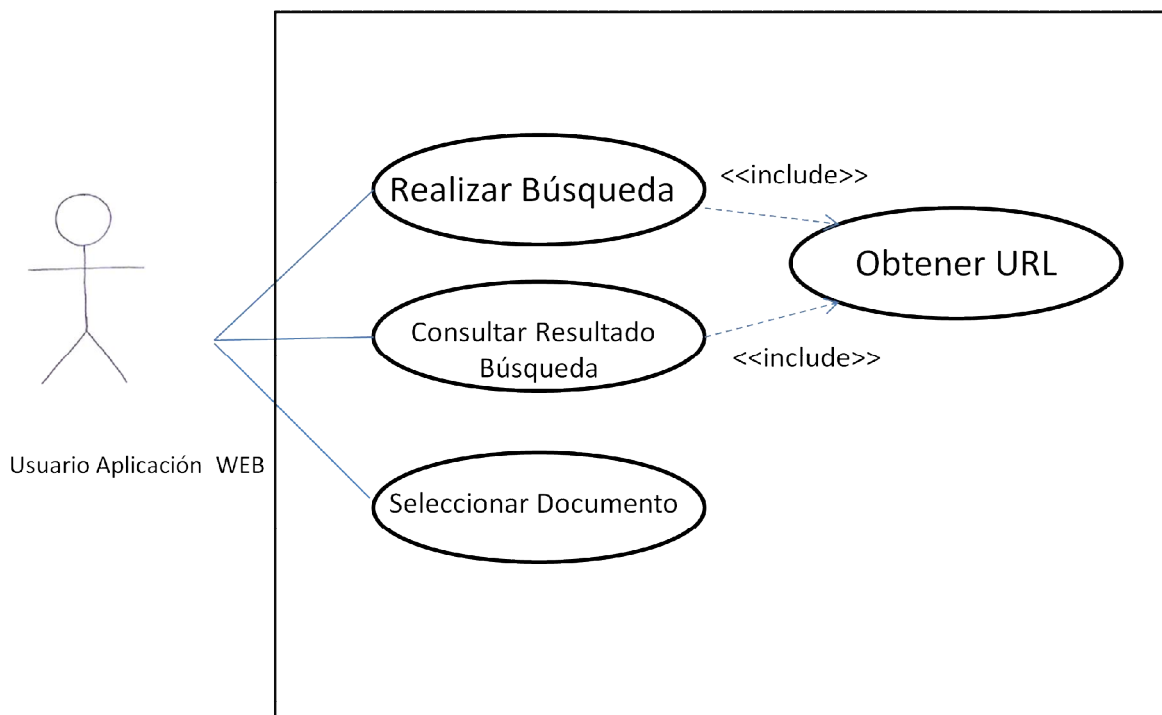


Ilustración 6: Caso de Uso Buscador

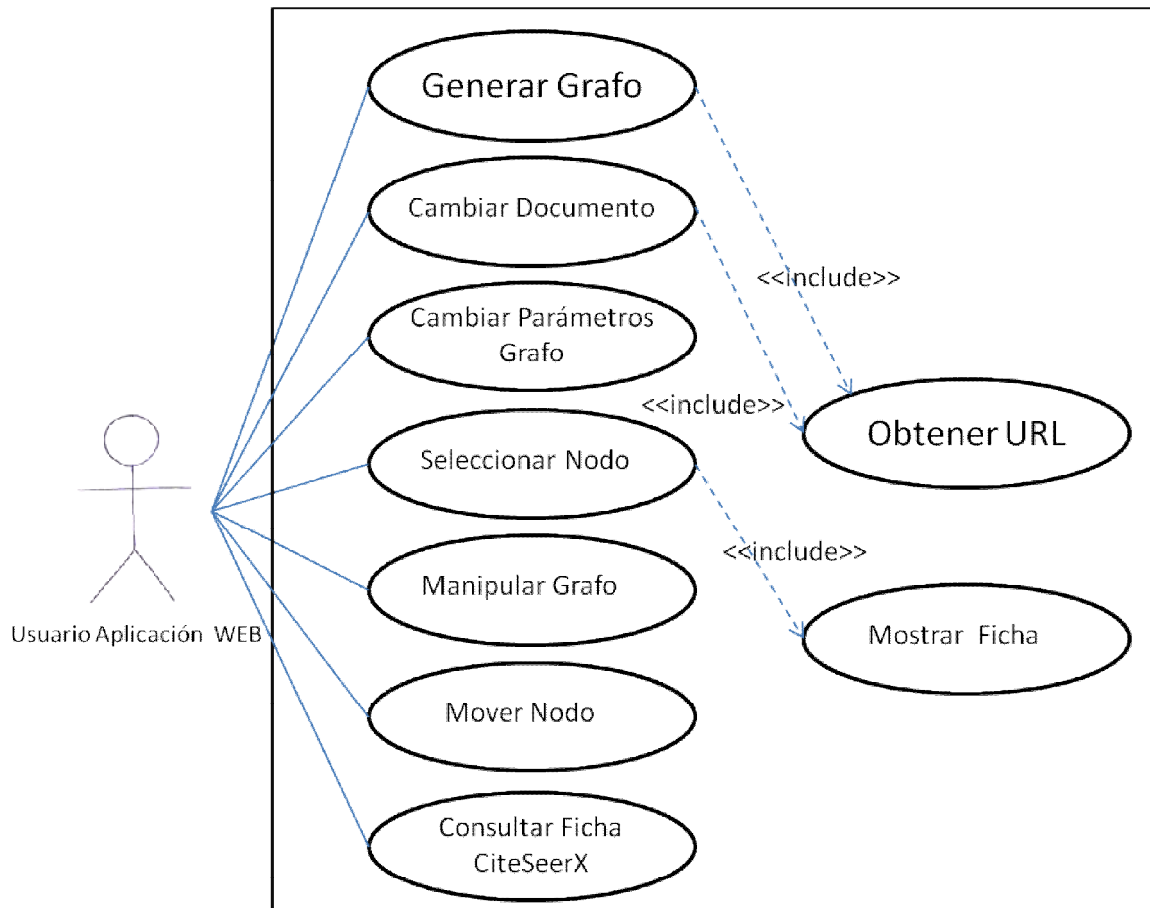


Ilustración 7: Caso de Usos Visor Grafo

3.1.2.2 Descripción textual

Para cada uno de los casos de uso identificados en el apartado anterior se realizará una descripción textual para aclarar las funcionalidades. Los campos que describen los casos de uso son los siguientes:

- **Identificador:** Nombre de manera unívoca un caso de uso. La nomenclatura a seguir será CU-XX siendo XX números comprendidos entre 0 y 9 que se irán incrementando consecutivamente.

- **Nombre:** Descripción breve del caso de uso.

* Tanto el campo *Identificador* como *Nombre* no estarán identificados en la tabla como campos sino que formarán parte del título de las mismas.

- **Actor:** Agente externo que interacciona con el sistema.

- **Objetivo:** Breve explicación del caso de uso.

- **Precondiciones:** Condiciones que deben cumplirse para poder ejecutar el caso de uso.
- **Postcondiciones:** Condiciones que se producen tras la ejecución del caso de uso.
- **Escenario básico:** Interacción más típica entre actor y sistema detallando la información y los cambios observados en el sistema.
- **Errores/ Alternativas:** Ejecución del caso de uso con condiciones de error o caminos de decisión distintos al básico.

CU-01	Realizar Búsqueda
Actor	Usuario Aplicación WEB
Objetivo	El usuario busca un documento con un texto libre
Precondiciones	Tener abierta la aplicación
Escenario básico	1.- Se accede a la página principal de la aplicación 2.- Se teclea el texto que se desea buscar 3.- Se pincha en buscar 4.- El sistema solicita los datos a CiteSeerX 5.- El sistema muestra el resultado
Errores/ Alternativas	5a.- Mensaje de Error si CiteSeerX no responde o no hay conexión. 5b.- Mensaje si no hay documentos en la búsqueda.
Postcondiciones	El usuario puede visualizar el resultado de la búsqueda del bloque de documentos 1 al 10. Si hay más de 9 resultados.

Tabla 29: Ficha Caso de Uso 01

CU-02	Resultado Búsqueda
Actor	Usuario Aplicación WEB
Objetivo	El usuario podrá seleccionar el siguiente bloque de 10 resultados o el anterior. Menos en el primer bloque o en el último
Precondiciones	Haber realizado una búsqueda con resultado y hay más de 10 líneas
Escenario básico	1.- El usuario quiere ver más resultados de su búsqueda 2.- El usuario pincha en nexto prevq 3.- El sistema solicita los datos a CiteSeerX 4.- El sistema muestra el resultado
Errores/ Alternativas	4a.- Mensaje de Error si CiteSeerX no responde o no hay conexión.
Postcondiciones	El usuario puede visualizar el resultado de la búsqueda del documento del bloque seleccionado

Tabla 30: Ficha Caso de Uso 02

CU-03	Seleccionar Documento
Actor	Usuario Aplicación WEB
Objetivo	El usuario selecciona un documento para generar su grafo
Precondiciones	Tener una lista de documentos
Escenario básico	1.- Tras una búsqueda con resultado 2.- Se selecciona un documento haciendo click sobre el título

Errores/ Alternativas	Nada
Postcondiciones	Se abrirá la ventana de generar grafo

Tabla 31: Ficha Caso de Uso 03

CU-04	Generar Grafo
Actor	Usuario Aplicación WEB
Objetivo	Generar el grafo según los parámetros indicados, con los datos obtenidos de CiteSeerX
Precondiciones	Haber seleccionado un documento
Escenario básico	1.- Se abre una nueva ventana 2.- Se obtiene la información para el grafo 3.- Se extrae la información 4.- Se genera el grafo 5.- Se genera el resumen de nodos y aristas
Errores/ Alternativas	2a.- Si no hay conexión se mostrar un mensaje de error 3a.- Si se encuentra algún error se mostrará un mensaje 4a.- Si no cumple el tipo de navegador mostrar un mensaje de aviso
Postcondiciones	Se mostrará un grafo con los datos obtenido

Tabla 32: Ficha Caso de Uso 04

CU-05	Cambiar Documento
Actor	Usuario Aplicación WEB
Objetivo	Generar el grafo del documento del nodo seleccionado
Precondiciones	Haber seleccionado un nodo
Escenario básico	1.- Se pincha en el enlace de Crear Grafo 2.- Se abre una nueva ventana para el nuevo grafo
Errores/ Alternativas	Nada
Postcondiciones	Se abre una nueva ventana con el grafo nuevo

Tabla 33: Ficha Caso de Uso 05

CU-06	Cambiar Parámetros Grafo
Actor	Usuario Aplicación WEB
Objetivo	Poder cambiar los parámetros de nivel y referencias mínimas para el grafo del documento seleccionado
Precondiciones	Tener creado un grafo
Escenario básico	1.- Modificar los parámetros 2.- Redibujar el grafo
Errores/ Alternativas	Nada
Postcondiciones	Un nuevo grafo del documento seleccionado

Tabla 34: Ficha Caso de Uso 06

CU-07	Seleccionar Nodo
Actor	Usuario Aplicación WEB
Objetivo	Visualizar la ficha del nodo seleccionado o mover el nodo dentro del grafo
Precondiciones	Tener un grafo dibujado
Escenario básico	1.- Se selecciona el nodo 2.- Se muestra la ficha

	3.- Se resaltan el nodo y las aristas del nodo
Errores/ Alternativas	Nada
Postcondiciones	Se visualiza la ficha del nodo seleccionado

Tabla 35: Ficha Caso de Uso 07

CU-08	Manipular Grafo
Actor	Usuario Aplicación WEB
Objetivo	Poder hacer zoom y mover el grafo
Precondiciones	Tener un grafo
Escenario básico	1.- Pulsar en el acción(zoom +, zoom -, mover derecha, mover izquierda, mover arriba, mover abajo) 2.- Realizar acción
Errores/ Alternativas	2a.- No hacer nada si se superó el limite
Postcondiciones	Una nueva visión del grafo

Tabla 36: Ficha Caso de Uso 08

CU-09	Mover Nodo
Actor	Usuario Aplicación WEB
Objetivo	Mover un nodo dentro del grafo
Precondiciones	Tener un grafo y un nodo seleccionado
Escenario básico	1.- Sin soltar el botón del ratón mover el nodo 2.- Soltar el botón del ratón al estar el nodo la nueva posición
Errores/ Alternativas	Nada
Postcondiciones	Una nueva posición del nodo en el grafo

Tabla 37: Ficha Caso de Uso 09

CU-10	Consultar Ficha CiteSeerX
Actor	Usuario Aplicación WEB
Objetivo	Visitar la Ficha del documento en CiteSeerX
Precondiciones	Haber seleccionado un nodo y tener activa la Ficha
Escenario básico	1.- Pinchar en el enlace
Errores/ Alternativas	Nada
Postcondiciones	Se abre una nueva ventana con la ficha en CiteSeerX

Tabla 38: Ficha Caso de Uso 10

3.1.3 Matriz de trazabilidad

Matriz de trazabilidad entre requisitos funcionales y casos de uso

Casos de Uso/Requisitos Funcionales	CU-01	CU-02	CU-03	CU-04	CU-05	CU-06	CU-07	CU-08	CU-09	CU-10
RF-01	X	X		X	X					
RF-02	X									
RF-03			X							
RF-04		X								
RF-05	X	X								
RF-06	X	X								
RF-07			X							
RF-08				X						
RF-09				X		X				
RF-10				X	X		X		X	
RF-11				X		X				
RF-12				X		X				
RF-13				X		X				
RF-14				X		X				
RF-15							X			
RF-16					X					
RF-17										X
RF-18				X		X				
RF-19								X		
RF-20								X		
RF-21									X	
RF-22		X								

Tabla 39: Matriz de Trazabilidad

3.2 Diseño

3.2.1 Arquitectura de la aplicación

En este punto vamos de describir la arquitectura que tendrá nuestra aplicación web. La aplicación web no obtendrá los datos de una Base de Datos directamente, sino que se proveerá de la información de la URL de CiteSeerX. Como podemos ver en la imagen el usuario no tiene por qué estar conectado a internet, siendo el servidor de publicación de la aplicación quien lo esté.

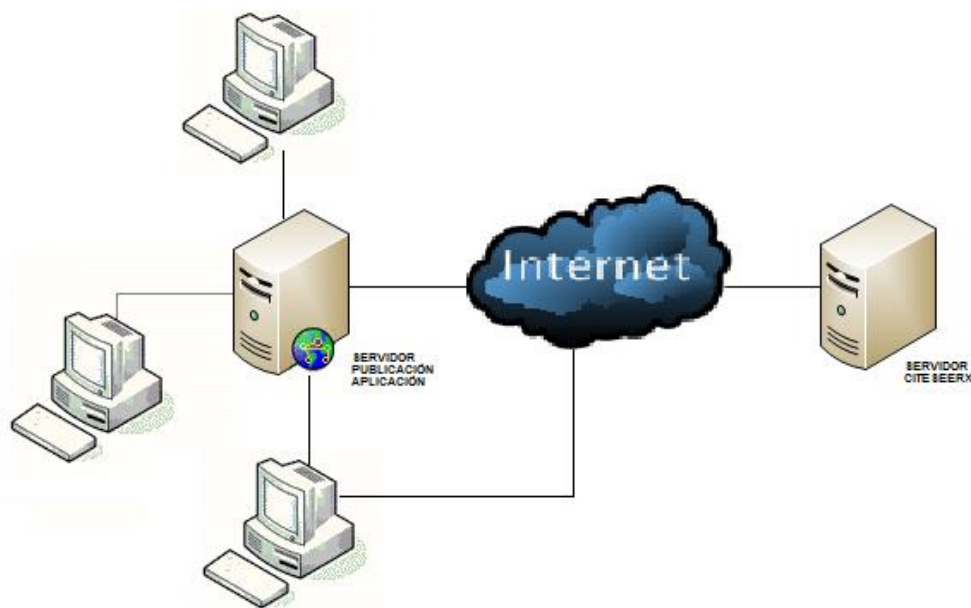


Ilustración 8: Arquitectura del Sistema

El usuario realizará una búsqueda y la aplicación en su lado de servidor (PHP) la convertirá en una URL que será lanzada a internet. Llegada la petición al servidor de CiteSeerX este devolverá el contenido de la URL al servidor de la aplicación. El servidor enviará el resultado al lado del cliente (JavaScript) que analizará y extraerá los datos necesarios para mostrar el resultado de la búsqueda. Más adelante explicaremos que estructura de datos hemos utilizado para almacenar los datos extraídos (ver punto 3.2.3.).

Este mismo proceso se realizará cuando se haya seleccionado un documento y se quiera generar un grafo. En este caso se realizarán tantas llamadas como documentos referencien al documento citado, para obtener la información de cada documento.

3.2.2 Diseño de la Aplicación web

Desde el punto de vista del diseño de la aplicación se ha buscado que cumpla una serie de características generales. Estas características generales se basan en los requisitos de usuario y fueron localizadas en la fase de análisis.

Se ha dividido en dos plantillas diferenciadas que además pudieran estar abiertas simultáneamente para facilitar así su uso.

- Plantilla de Búsqueda
- Plantilla del Grafo.

3.2.2.1 Diagrama de Navegación

En la siguiente figura se puede observar el diagrama de navegación de la aplicación realizada, de este modo se puede conocer con facilidad el número de ventanas de las que consta nuestra aplicación así como los pasos necesarios para conseguir el fin deseado.

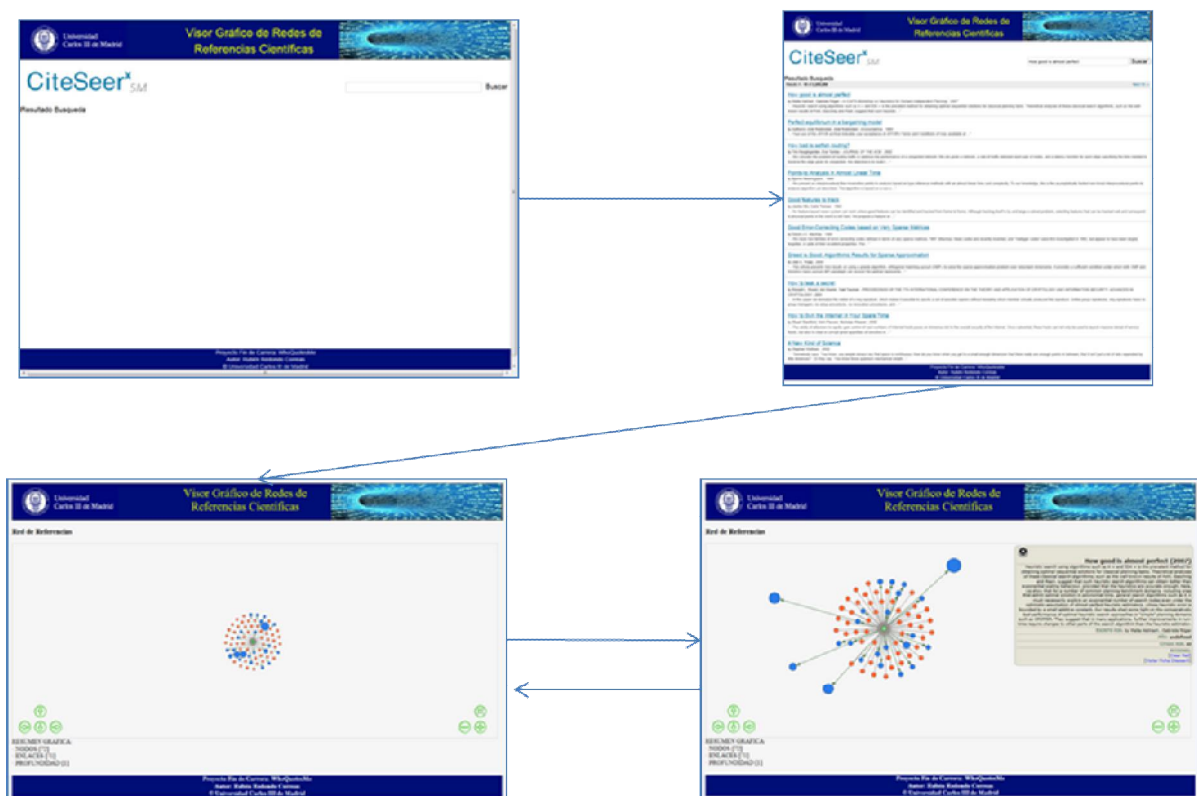


Ilustración 9: Diagrama de navegación

3.2.2.2 Plantillas Aplicación web

Ambas plantillas comparten una serie de características similares.

Para facilitar el uso de la aplicación se ha creado unas plantillas intuitivas y simples. Alejándose de acciones y efectos decorativos y sin funcionalidad real.

Se le ha dado un aspecto corporativo con un toque personal. Se ha diseñado una bloque cabecera y bloque pie igual en ambas plantillas con los colores y los iconos de la Universidad Carlos III de Madrid. Se ha simplificado la información contenido en ambos elementos, incluyendo un gráfico que representa la entrada a la red de internet.



Ilustración 10: Cabecera Común de la Aplicación web

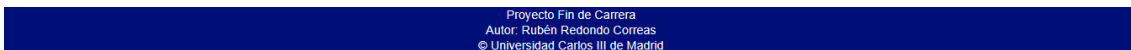


Ilustración 11: Pie Común de la Aplicación web

Como las plantillas son independientes se podrá tener abierta varios grafos para poder compararlos.

Cada vez que se realizase una acción que requiera de la espera por parte del usuario se bloqueará la pantalla.

Plantilla de Búsqueda

La plantilla de búsqueda pretende ser parecida al buscador de la página de CiteSeerX para comodidad del usuario. Por eso se ha incorporado la imagen de la web. Los resultados de la búsqueda muestran un aspecto similar al de CiteSeerX.

A continuación se describen las partes principales de la plantilla de búsqueda.

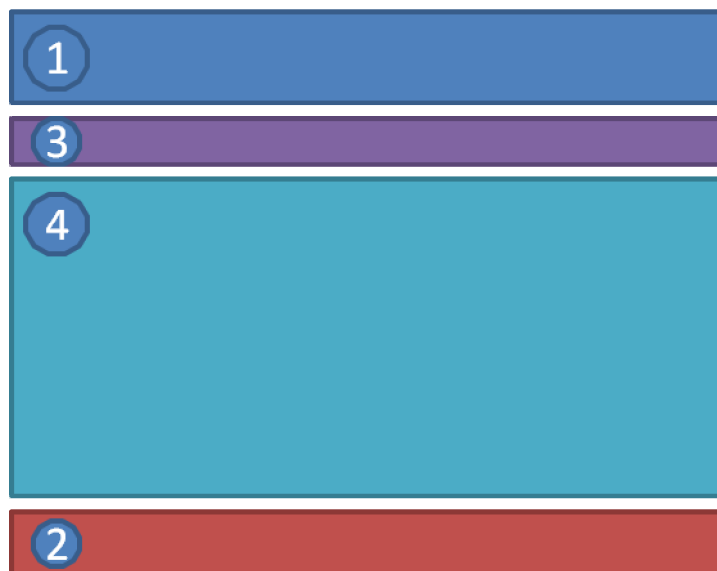


Ilustración 12: Esquema Plantilla Búsqueda

1. Cabecera de la plantilla. Contiene el logo de la U. Carlos III, el nombre de la aplicación y una imagen decorativa.
2. Pie de la plantilla. Contiene información general sobre la aplicación web.
3. Zona de búsqueda. Aquí se encuentran los elementos para realizar una búsqueda, caja de texto y botón de buscar.
4. Aquí se mostrará el resultado de la búsqueda, y los errores en caso de producirse. Se mostrará el total de resultados obtenidos así como el bloque mostrado. También se encuentra la zona de navegación con el bloque siguiente y bloque anterior.

Plantilla del Grafo

En esta plantilla se busca que el grafo tenga un protagonismo directo y por ello la ficha sale oculta inicialmente.

A continuación se describen las partes principales de la plantilla del grafo.

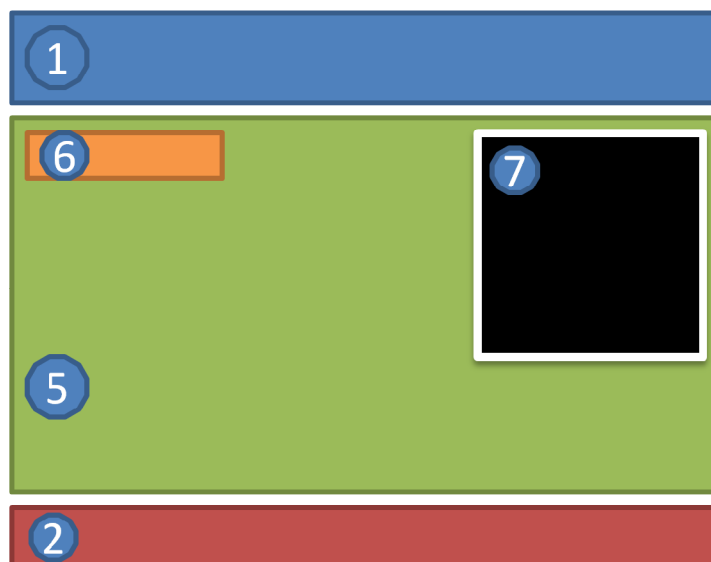


Ilustración 13: Esquema Plantilla Grafo

1. Cabecera de la plantilla. Contiene el logo de la U. Carlos III, el nombre de la aplicación y una imagen decorativa.
2. Pie de la plantilla. Contiene información general sobre la aplicación web.
5. Zona del Grafo. Aquí se encuentra el grafo y sus elementos de navegación: hacer zoom y mover el grafo.
6. Resumen del Grafo. Aquí se mostrará un resumen de los elementos del grafo, nodos, aristas, nivel de búsqueda. Y se podrá modificar los parámetros visuales del grafo.
7. Ficha Documento. Esta es el cuadro de la ficha de la información del documento seleccionado. Al seleccionar un documento (nodo) se mostrará su información: título, autor, año publicación, número de documentos que le citan. También se verán las acciones posibles a realizar sobre el documento: generar su grafo o ver la ficha en la web de CiteSeerX.

3.2.3 Estructura de Datos

La aplicación web no realiza llamadas a ninguna base de datos si no que extrae la información del contenido HTML de la URL de CiteSeerX. En nuestro caso la URL es nuestro proveedor de datos. A continuación vamos de explicar que estructuras de datos hemos utilizado para almacenar en memoria la información de la estructura de datos del grafo y la información de cada nodo o arista.

Tipo de Dato NODO

CAMPO	TIPO	DESCRIPCIÓN
cid	string	ID interno con la lista de documentos que le referencia
nciti	int	Numero de documentos que lo referencian
titulo	string	Título del documento
autor	string	Autor del documento
anno	string	Año del documento
nota	string	Resumen del contenido del documento
param_idoc	string	ID del documento, es un estándar usado en otras web
citado	boolean	Si es referencia o referencia del documento seleccionado.

Tabla 40: Tipo de Dato NODO

Tipo de Dato ARISTA

CAMPO	TIPO	DESCRIPCIÓN
from	string	cid del nodo origen
to	string	cid del nodo destino

Tabla 41: Tipo de Dato ARISTA

Tipo de Datos Grafo

CAMPO	TIPO	DESCRIPCIÓN
Nodos	Array Nodo	Lista de Nodos del Grafo
Aristas	Array Arista	Lista de Aristas del Grafo

Tabla 42: Tipo de Datos Grafo

Ahora se explica cómo se realiza la llamada a la URL de CiteSeerX para obtener la información y así poder cargar la información en la estructura que se explicó antes.

<http://citeseerx.ist.psu.edu>

URL de Búsqueda: Permite realizar una búsqueda.

- <http://citeseerx.ist.psu.edu/search>
 - submitoperación [Search]
 - q texto de la búsqueda
 - sort ordenación [rlv, cite, date, ascdatetime, recent]
 - t tipo de fichero[doc]
 - start nº inicial del bloque de 10 documentos

URL de ficha de documento: Permite visualizar la ficha de un documento

- <http://citeseerx.ist.psu.edu/viewdoc/summary>
 - doi numero ID de documento
 - cid numero ID de citación

URL de lista de documentos: permite ver un bloque (10 elementos) de documentos que referencian a uno.

- <http://citeseerx.ist.psu.edu/showciting>
 - cid numero ID de citación
 - sort ordenación [date, recent, cite, ascdatetime]

3.3 Implementación

Los lenguajes de programación elegidos para la implementación del proyecto de fin de carrera son HTML5, CSS3 y JavaScript. Como apoyo en el JavaScript se ha utilizado el Framework jQuery y una librería gráfica VISJS.

HTML es un código estándar que es manejado por cualquier navegador y es independiente de la plataforma sobre la que se ejecute. HTML5 es la última versión del importante código HTML y dispone de nuevas herramientas que hacen más cómoda la programación. Incorpora la etiqueta `<CANVAS>` que nos permite insertar objetos gráficos.

Se ha elegido CSS3 por ser la implementación mediante hojas de estilo en cascada, manteniendo así una separación entre la estructura de la herramienta y su presentación.

Para agilizar y hacer compatible en todos los navegadores se ha incorporado el Framework jQuery que permite dinamizar la tareas en JavaScript. Permitiendo extraer la información recibida de CiteSeerX de una forma más sencilla.

En la parte de representación gráfica se optó por una librería sencilla llamada VISJS que nos permite crear un grafo de forma sencilla. Pudiendo gestionar el aspecto del grafo de forma rápida, así como interactuar con sus eventos.

Para solicitar a través de una llamada AJAX el contenido de la web de CiteSeerX ha sido PHP.

La herramienta de desarrollo usada ha sido Microsoft WEBMatrix v3.0 que permite ejecutar código PHP. Y el propio depurador del IE para el código JavaScript.

Para poder probar de una forma más autónoma la aplicación se ha impreso la funcionalidad de cargar un grafo desde los datos de un fichero.

A continuación se describirá la estructura de ficheros y se explicarán los procesos más importantes en la implementación de la aplicación web.

css

- cites-network.css
- frame.css
- main.css
- reset.css
- results.css

img

js

- Archivo.js
- cite-network.js
- DatosGrafo.js
- jquery-1.11.2.min.js
- jquery-2.1.3.min.js
- jquery.blockUI.js
- jquery.blockUI207.js

vis

-  **img**
- vis.css
- vis.js
- vis.map
- vis.min.css
- vis.min.js
- **ba-simple-proxy.php**
- **GlobalValor.js**
- **Grafo.html**
- **Index.html**

Ilustración 14: Árbol de ficheros de la Aplicación

3.3.1 Ficheros

La aplicación está formada por varios paquetes que representan los grupos de acciones similares en el contexto de la aplicación. Todos los ficheros están codificados con la tabla de caracteres `%UTF-8+`.

Como es habitual en las implementaciones de código, cuando es necesario, el código se encuentra comentado y "*referenciado*" en lo relativo a restricciones funcionales y no funcionales, con nombres de variables explicativos y métodos cuya nomenclatura es explicativa en sí misma y "*camel sensitive*"⁸.

A continuación describiremos los ficheros más importantes que tiene el proyecto, explicando los métodos que se han considerado más relevantes.

3.3.1.1 *Index.html*

Es el fichero principal de la aplicación, toma su nombre del estándar de ejecución de los servidores de aplicaciones. Es una de las plantillas de la aplicación. Se encarga de lanzar las búsquedas, así como la plantilla de la generación del grafo.

- *Load()* - Sin comentario.
- *buscar()* . Realiza la llamada al fichero Proxy (`ba-simple-proxy.php`), extrae del código HTML recibido el resultado de la búsqueda. Elimina el código HTML sobrante y añade el enlace al bloque anterior.

3.3.1.2 *Grafo.html*

Es la plantilla que genera el grafo. Es desde la que se puede realizar la manipulación del grafo, así como ver la ficha de un documento.

- *Load()* . Se encarga de solicitar la información para el grafo.
- *tomarAccion()* . Lanza la verificación de requisito del navegador. Y en caso positivo carga el fichero para la creación del grafo.

3.3.1.3 *ba-simple-proxy.php*

Es el fichero que contiene el código PHP que realiza la función de proxy. Gracias a este código se puede traer el contenido de una URL que no esté en tu propio dominio. Es el único código que se ejecuta en el lado del servidor.

⁸ *camel sensitive*: nomenclatura utilizada en lenguajes como Java para los métodos; una frase explicativa de su función, sin espacios y con la primera letra de cada palabra en mayúscula; ejemplo: `DamIdentificadorDelObjeto`

3.3.1.4 GlobalValor.js

En este fichero se encuentran las constantes globales de la aplicación y todas las funciones que son generales.

- *PROXY* = *'ba-simple-PROXY.php'*; - El fichero de proxy.
- *URL_MASTER* = *'http%3A%2F%2Fciteeex.ist.psu.edu'*; - URL del buscador usado⁹.
- *PATH_FICHA* = *'%2Fviewdoc%2Fsummary'*; - Path en la URL de la ficha en el buscador.
- *PATH_BUSQUEDA* = *'%2Fsearch%3Fq%3D'*; - Path en la URL de búsqueda en el buscador.
- *PATH_CITAS* = *'%2Fshowciting'*; - Path en la URL de la lista de citas en el buscador.
- *VAR_IDDOC* = *'doi'*; - Nombre del parámetro en la URL del doi en el buscador.
- *VAR_IDCITA* = *'cid'*; - Nombre del parámetro en la URL del cid en el buscador.
- *NIV_MAX* = *1*; - Nivel de profundidad de la búsqueda.
- *NUM_LINEAS* = *10*; Número de documentos en un bloque de resultado de búsqueda.
- *WEB* = *'S'*; Define si los datos se cargan desde la URL de búsqueda o un fichero.
- *getGET(url)* Devuelve la lista de parámetros de una URL.

3.3.1.5 cite-network.js

En este fichero se encuentran todas las funciones necesarias para mostrar el grafo. Así como todas las acciones posteriores.

- *graficosNotOk()* - Esta función verifica si se cumple las características del navegador para ver los gráficos CANVAS, si no es así muestra un mensaje de aviso.
- *getNodeInfo(n)* - Genera el código HTML con la información del nodo seleccionado. Este código se introducirá en el <div> de la ficha.
- *MostrarPanelInfo(nodo)* - Oculta o hace visible el <div> de la ficha.
- *verColorNodo(valor, cid)* - Define el color de un nodo.
- *calcularScala(valor, maximo)* . Calcula el tamaño del nodo en una escala de 1 a 10.
- *pintarResumen(lista_nodos, lista_nodos_informacion)* . Pinta la ficha resumen del grafo.
- *graficosOk()* - Si se cumple que el navegador puede mostrar gráficos CANVAS, carga los datos del grafo desde el objeto grafo, da propiedades al grafo y lo visualiza.

⁹ El buscador usado es CITESEERX

```

Cargamos los nodos del grafo
    Calculamos color del nodo
    Calculamos tamaño del nodo
Cargamos las aristas del grafo
Definimos propiedades del grafo, nodos y aristas
Definimos el evento "select" del grafo
Cargamos el grafo

```

3.3.1.6 DatosGrafo.js

En este fichero se encuentran las funciones necesarias para la extracción de la información previa a la creación del grafo. Tanto pidiendo URLs al proxy, así como extrayendo la información recibida.

- *sacarPadresHijos(micid, nivel)* . Se considera que los padres son los documentos que le referencia. Y los hijos aquellos a los que hace referencia.
- *sacarHijos(midoi, nivel)* . Saca de una única llamada al proxy, por la ficha del documento, la lista
- *sacarPadres(micid, nivel)* . Realizando llamadas al proxy obtiene la información de los documentos padre. Las llamadas van en bloque de 10 documentos.
- *recuperarContents(url)* Hace la llamada al proxy a partir de una URL.
- *sacarAnnoHijo(texto)* - Sin comentario.
- *limpiarCitas(texto_citi)* - Sin comentario.
- *existenPaginas(texto)* - Sin comentario.
- *añadirElementoActual(micid, num_citis, text_contents)* - Sin comentario.
- *calcularEnlacePagAnte()* . Añade al código HTML el enlace al bloque anterior.

A continuación se va a explicar el proceso de obtención de la información desde la web CiteSeerX. Este se realiza de forma recursiva¹⁰.

```

Parámetros:  cid      código del documento
              nivel    nivel del profundidad de obtención

Proceso sacarPadresHijos(micid, nivel)
    Si nivel es <= NIVEL_MAX
        Migrafo = Migrafo + sacarHijos(midoi, nivel)
        Migrafo = Migrafo + sacarPadres(micid, nivel)

```

¹⁰ **recursividad** es la forma en la cual se especifica un proceso basado en su propia definición.


```

    Fin Si
    Retorno Migrafo
Fin Proceso

Proceso sacarHijos(micid, nivel)
    Llamada al proxy
    Mientras hay hijos
        Obtener info hijo
        Añadir info hijolista
        ListaHijos = ListaHijos + hijolista
    Fin Mientras
    Retorno ListaHijos
Fin Proceso

Proceso sacarPadres(micid, nivel)
    Hacer
        Llamada al proxy bloqueN
        Mientras tenga padres el bloqueN
            Obtener info padre
            Añadir info padrelista
            Listapadre = Listapadre + padreista
            sacarPadresHijos(padrecid, nivel + 1)
        Fin Mientras
        bloqueN = bloqueN + 10
    Mientras haya Padres
Fin Proceso

```

3.3.1.7 Archivo.js

Aquí se encuentra la función que permite cargar un fichero de texto para suplir a la llamada al proxy.

- *cargarFichero(file)* - Coge como parámetro un nombre de fichero que carga en el objeto Grafo.

3.4 Pruebas

Una vez creada la aplicación web se realizará las pruebas para verificar su correcto funcionamiento. A continuación se mostrará un resumen de la batería de pruebas realizada a la aplicación web una vez finalizada la fase de implementación.

Durante la fase de implementación de la aplicación web se han desarrollado pruebas de procesos, detectando errores que se han corregido.

A pesar de haber realizado estas pruebas durante la implementación se ha creado una batería de pruebas para poder verificar que se hayan cumplido todos los requisitos expuestos en la fase de análisis.

Para poder realizar pruebas sin tener conexión a internet, ni acceso a la web de CiteSeerX, se ha generado un pequeño código, ya comentado en la parte de implementación, que nos ayuda a cargar los datos para la generación del grafo desde un fichero de texto. Este desarrollo no es parte de los requisitos pero ha ayudado a localizar de una forma más cómoda posibles problemas o deficiencias en la obtención de la información.

Se ha realizado unas pruebas de visualización en distintos navegadores y sobre distintas resoluciones de pantalla.

También se le ha realizado una prueba de esfuerzo a la parte de obtención y creación del grafo.

Todas las pruebas se realizaron con la aplicación web en un alojamiento básico de Hostalia¹¹, y un pc conectado a internet con fibra de 20Mb. Las características del PC fueron:

- HP ProBook 4530s
- Inter® Core i5-2450M CPU 2.5GHz
- 4,00 GB Ram
- Windows 7 64bits
- 1 GB HDD

A nivel general y una vez realizadas las pruebas podemos decir que la aplicación web cumple de forma satisfactoria las especificaciones solicitadas.

¹¹ <http://www.hostalia.com/>

3.4.1 Prueba Visualización

Aunque la aplicación está desarrollada en una resolución de 1280x960 se ya verificado que funciona correctamente para otras resoluciones, a continuación tenemos una tabla donde se indican los resultados de estas pruebas sobre distintos navegadores. Gracias al editor WebMatrix hemos podido lanzar de forma cómoda la aplicación web en distintos navegadores previamente instalados.

	Internet Explore	Mozilla FireFox	Google Chrome	Safari	Opera
1024x768	Pasado	Pasado	Pasado	Pasado	Pasado
1280x720	Pasado	Pasado	Pasado	Pasado	Pasado
1280x960	Pasado	Pasado	Pasado	Pasado	Pasado
1600x900	Pasado	Pasado	Pasado	Pasado	Pasado
1920x1080	Pasado	Pasado	Pasado	Pasado	Pasado
2560x1600	Pasado	Pasado	Pasado	Pasado	Pasado

Tabla 43: Prueba de Visualización

3.4.2 Pruebas de Requisitos

Vamos a explicar las pruebas realizadas para poder verificar los requisitos de la aplicación. Estas pruebas se exponen de forma tabular en el cual se incluyen los siguientes contenidos para cada una de ellas:

ID: Indica de manera unívoca a una prueba.

DESCRIPCIÓN: especificación de la prueba.

NIVEL DE ERROR: resultado de la prueba. Cuyos valores son: 1 pasada, 2 pasada en ocasiones, 3 no pasada.

ID	PR_01
DESCRIPCIÓN	Se realiza una búsqueda con un texto que va a devolver al menos un resultado.
NIVEL DE ERROR	1

Tabla 44: Prueba PR_01

ID	PR_02
DESCRIPCIÓN	Se realiza una búsqueda con un texto que no va a devolver ningún resultado.
NIVEL DE ERROR	1

Tabla 45: Prueba PR_02

ID	PR_03
DESCRIPCIÓN	Se realiza una búsqueda sin texto.
NIVEL DE ERROR	1

Tabla 46: Prueba PR_03

ID	PR_04
DESCRIPCIÓN	Se verifica página next y prev. En la 1ª, última y una de en medio.
NIVEL DE ERROR	1

Tabla 47: Prueba PR_04

ID	PR_05
DESCRIPCIÓN	Mover nodo de posición
NIVEL DE ERROR	1

Tabla 48: Prueba PR_05

ID	PR_06
DESCRIPCIÓN	Seleccionar nodo y visualizar ficha. Cambiar de nodo seleccionado.
NIVEL DE ERROR	1

Tabla 49: Prueba PR_06

ID	PR_07
DESCRIPCIÓN	Crear nuevo grafo.
NIVEL DE ERROR	1

Tabla 50: Prueba PR_07

ID	PR_08
DESCRIPCIÓN	Verificar en IE < 9
NIVEL DE ERROR	1

Tabla 51: Prueba PR_08

ID	PR_09
DESCRIPCIÓN	Hacer zoom y mover el grafo.
NIVEL DE ERROR	1

Tabla 52: Prueba PR_09

ID	PR_10
DESCRIPCIÓN	Visualizar resumen grafo
NIVEL DE ERROR	1

Tabla 53: Prueba PR_10

ID	PR_11
DESCRIPCIÓN	Sobre un documento %How good is almost perfect+ sabiendo cual es su grafo, verificar que es correcto.
NIVEL DE ERROR	1

Tabla 54: Prueba PR_11

ID	PR_12
DESCRIPCIÓN	Desde la ficha poder acceder a la ficha de CiteSeerX
NIVEL DE ERROR	2

Tabla 55: Prueba PR_12

Tabla de cruce prueba y requisitos

Requisitos	PR_01	PR_02	PR_03	PR_04	PR_05	PR_06	PR_07	PR_08	PR_09	PR_10	PR_11	PR_12
RF-01	X	X	X									
RF-02	X	X	X									
RF-03					X	X	X		X	X		
RF-04				X								
RF-05				X								
RF-06	X											
RF-07					X	X	X		X	X		
RF-08	X											
RF-09											X	
RF-10					X	X						
RF-11					X	X						
RF-12					X	X						
RF-13					X	X						
RF-14					X	X						
RF-15						X						
RF-16							X					
RF-17												X
RF-18										X		
RF-19									X			
RF-20									X			
RF-21								X	X			
RF-22				X								

Tabla 56: Tabla de cruce prueba y requisitos

3.4.3 Pruebas de Esfuerzo

Sobre una escala de valores de 5 a 1, siendo 5 lo óptimo y 1 lo peor, se valora la ejecución y manipulación del grafo por su número de nodos y aristas. Para poder realizar esta prueba se generó una herramienta para generar sintéticamente grafos de un número de nodos y aristas deseado y así poder valorar su rendimiento.

Nodos/Aristas	0-20	21-50	51-100	101-200	201-500	500+
0-20	5	5	-	-	-	-
21-50	5	5	5	-	-	-
51-100	5	5	5	5	5	5
101-200	5	5	5	5	4	4
201-500	4	4	4	4	4	3
500+	3	3	2	2	2	2

Tabla 57: Pruebas de Esfuerzo

Resumiendo las pruebas realizadas en sus distintos ámbitos fueron satisfactorias. Y solo bajo el rendimiento ante grafos muy poblados.

4 Planificación

4.1 Planificación

4.1.1 Modelo del ciclo de vida

El ciclo de vida ha marcado la metodología de trabajo, de principio a fin, del proyecto y la planificación del mismo. El ciclo de vida del desarrollo software es una descripción de las distintas formas de desarrollo de un proyecto, es decir, la orientación que debe seguirse para obtener, a partir de los requisitos del cliente, sistemas que puedan ser utilizados por éste. Este término también se conoce como conjunto de fases, procesos y actividades requeridas para ofertar, desarrollar, probar, integrar, explotar y mantener un producto software. El modelo de desarrollo utilizado para la realización de la aplicación ha sido el modelo Desarrollo Rápido de Aplicaciones.

El Desarrollo Rápido de Aplicaciones, abreviado como RAD¹² es un modelo de ciclo de vida que enfatiza un desarrollo extremadamente corto. Se trata de una adaptación del modelo tradicional en cascada (Royce en 1970) en el que se logra el desarrollo rápido utilizando una construcción basada en componentes. Si se comprenden bien los requisitos y se limita el ámbito del proyecto, el proceso RAD permite crear un sistema completamente funcional dentro de periodos cortos de tiempo (entre 60 y 90 días).

En aplicaciones de sistemas de información, el enfoque RAD tiene las siguientes fases:

1. Modelado de Gestión: se modela el flujo de información entre las funciones de gestión.
2. Modelado de Datos: se refina el flujo de información como un conjunto de objetos de datos necesarios para apoyar la empresa. Se definen las características de cada uno de los objetos y sus relaciones.
3. Modelado del Proceso: se definen las transformaciones (añadir, modificar, suprimir o recuperar) sobre los objetos del modelo de datos para lograr los flujos de información de cada función de gestión.
4. Generación de Aplicaciones: codificación de una función de gestión.
5. Pruebas y entrega: prueba de los componentes y entrega del programa que realiza una función de gestión.

La clave del modelo RAD está en los cambios en las etapas de codificación y pruebas:

“ Codificación. El modelo RAD asume la utilización de técnicas de cuarta generación. En lugar de crear software con lenguajes de programación de tercera generación, se

¹² Rapid Application Development

trabaja para volver a utilizar componentes de programas ya existentes (cuando es posible) o a crear componentes reutilizables (cuando sea necesario). En todos los casos se utilizan herramientas para facilitar la construcción de software.

” Pruebas. Como se enfatiza la reutilización, ya se han comprobado muchos de los componentes de los programas. Esto reduce en muchos casos el tiempo de pruebas, aunque se deben probar todos los componentes nuevos y se deben ejercitar todas las interfaces a fondo.

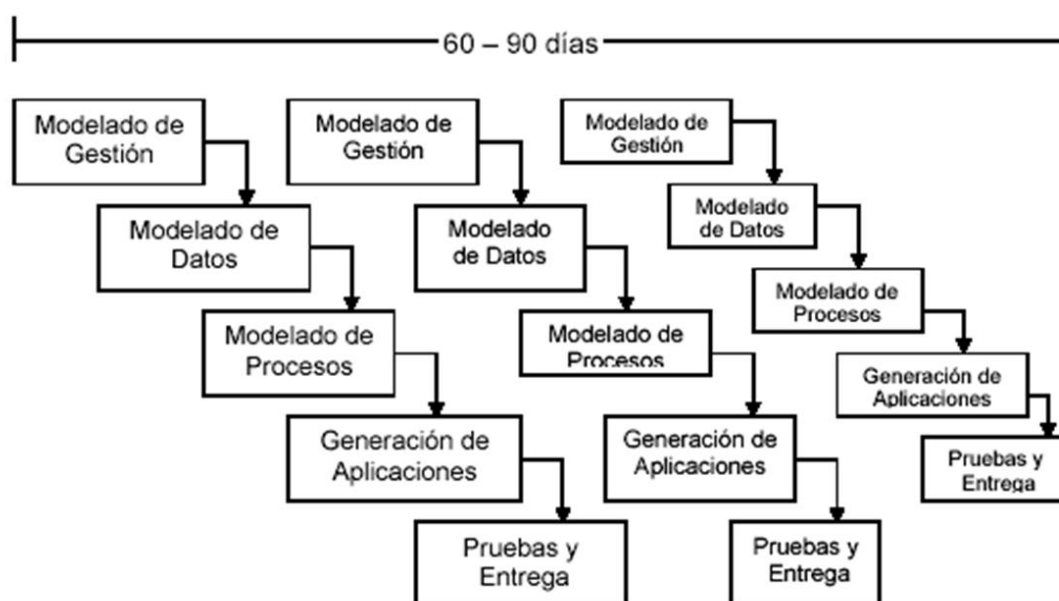


Ilustración 15: El modelo de ciclo de Vida RAD

Como nuestro proyecto era fácilmente modulable ya que tiene dos funcionalidades claramente identificadas, como son la búsqueda en CiteSeerX y la creación del grafo, ha permitido afrontar cada una de ellas por un equipo RAD. Y cada una de estas dos funcionalidades tiene a su vez unas sub-funcionalidades también claramente identificadas que han sido desarrolladas por otros equipos RAD. Todos los equipos RAD han seguido las distintas fases del modelo de ciclo de vida, punto 1-5 anteriormente explicados, independientemente unas de otras. Finalmente se ha integrado en un único conjunto las funcionalidades de los equipos RAD. Gracias al trabajo de los equipos RAD se ha conseguido desarrollar la aplicación web de una forma más rápida y se ha podido cubrir de forma óptima las necesidades del cliente.

Cada equipo RAD ha desarrollado pequeñas funcionalidades que han sido reutilizadas por los otros equipos ahorrando tiempos y costes.

4.1.2 Diagrama de Gantt

Una vez explicados los aspectos generales de la gestión del proyecto, procedemos a detallar la planificación del mismo. Para realizar la planificación se ha utilizado una herramienta básica en la rama de ingeniería del software; el diagrama de Gantt.

El diagrama de Gantt es una útil herramienta gráfica cuyo objetivo es exponer el tiempo de dedicación previsto para diferentes tareas o actividades a lo largo de un tiempo total determinado. A pesar de esto, el *Diagrama de Gantt* no indica las relaciones existentes entre actividades. Aunque la posición de cada tarea a lo largo del tiempo hace que se puedan identificar dichas relaciones e independencias. En gestión de proyectos, el diagrama de Gantt se ha convertido en una herramienta básica con la finalidad de representar las diferentes unidades mínimas de trabajo y las fases, tareas y actividades programadas como parte de un proyecto.

En la siguiente página se muestra la figura que contiene la planificación del proyecto completo. Teniendo en cuenta que cada día de trabajo corresponden a cuatro horas reales, excepto en la fase de documentación que corresponde a dos horas de trabajo. Podemos decir que el proyecto realizado ha tenido una duración de 90 días, 382 horas, donde se han realizado las fases de diseño, análisis, implementación, pruebas y documentación.

Como aspectos a destacar, resaltar las diferentes tareas de diseño que han sido realizadas en paralelo para reducir tiempo y recursos.

La tarea que ha durado más tiempo ha sido la implementación puesto que se encontró con la dificultad de la obtención de los datos

El calendario laboral utilizado para la planificación es un calendario típico donde los periodos laborables se encuentran de lunes a viernes y los periodos de descanso son los sábados y domingos. Además de los fines de semana encontramos con dos festivos. Estos festivos no influyen en las horas totales ya que se compensaron con dos sábados.

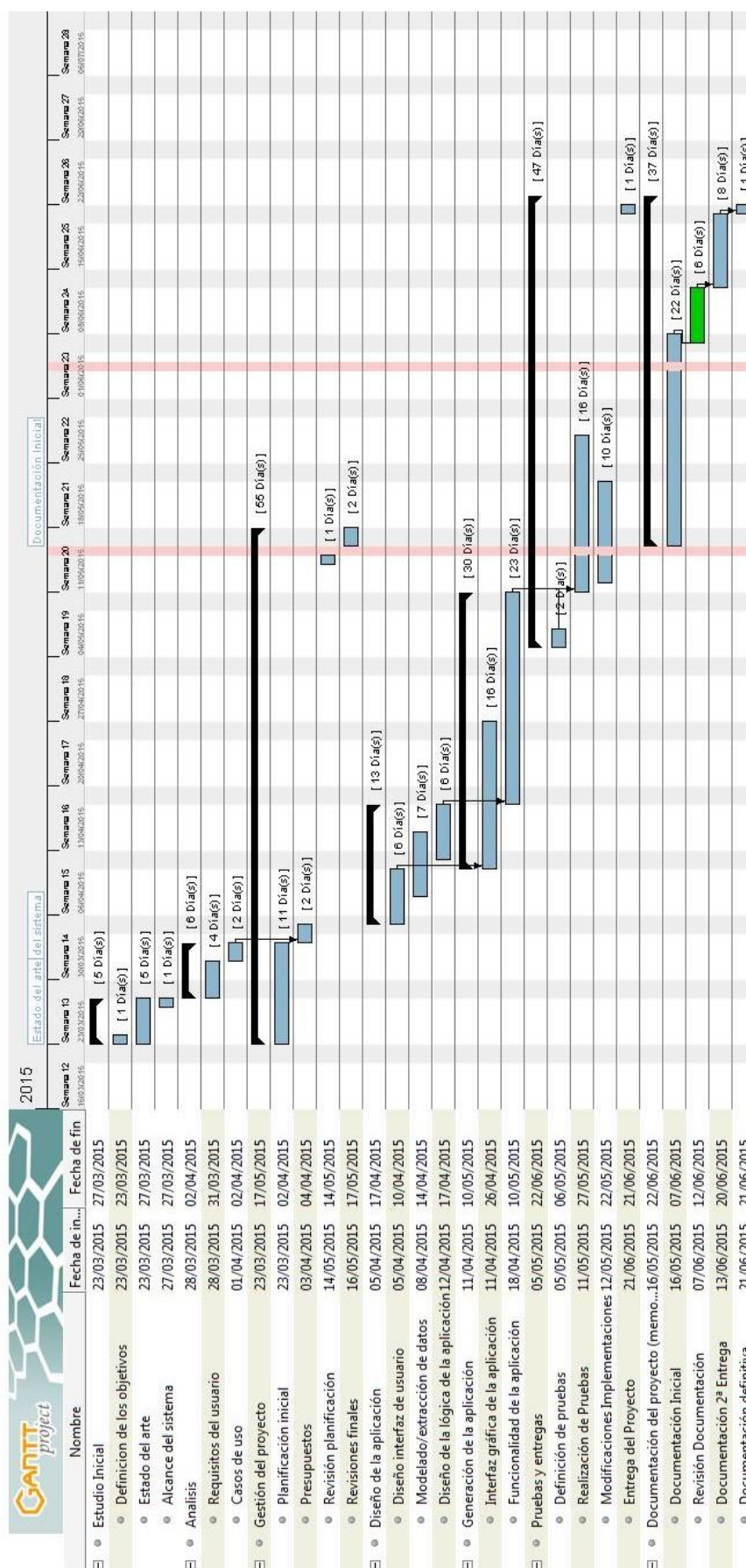


Tabla 58: Planificación del Proyecto sobre Diagrama de Gantt

4.2 Presupuesto

En este apartado se va a mostrar de manera detallada el presupuesto desglosado del proyecto, especificando los diferentes gastos que han sido necesarios para su realización.

4.2.1 Resumen de horas dedicadas

Basándonos en el diagrama de Gantt expuesto en apartados anteriores es posible calcular el número de horas totales dedicadas al proyecto.

Fase de análisis: 6 días x 4 horas = 24 horas

Fase de diseño: 13 días x 4 horas = 52 horas

Fase de implementación: 40 días x 4 horas = 160 horas

Fase de pruebas: 18 días x 4 hora = 72 horas

Fase de documentación: 37 días x 2 hora = 74 horas

Por tanto el número de horas totales dedicadas al proyecto será la suma de las horas dedicadas a cada una de las fases. El coste en horas de la totalidad del proyecto es de 382 horas.

4.2.2 Resumen de personal

En la siguiente tabla se muestran los cargos correspondientes al personal informático cualificado para realizar las distintas tareas o actividades. Los salarios por hora están en consonancia con los salarios de empleados en empresas similares del sector. Todos los costes son calculados sin I.V.A.

Cargo	Nº de horas	Coste Hora	Dedicación (hombre /mes)*	Total (Ö)
Diseñador	24	23,70 "	0,30	568,80 "
Ingeniero Sénior	52	39,50 "	0,65	2.054,00 "
Ingeniero	160	31,60 "	2,00	5.056,00 "
Responsable de pruebas	72	19,75 "	0,90	1.422,00 "
Responsable de documentación	74	19,75 "	0,93	1.461,50 "
TOTAL				10.562,50 Ö

Tabla 59: Resumen de personal

*Utilizando 1 Hombre/ mes = 80 horas según la planificación realizada

4.2.3 Resumen de Hardware

En la siguiente tabla se muestran los equipos informáticos adquiridos con su coste de amortización durante el periodo que dura el proyecto. Todos los costes son calculados sin I.V.A

Descripción	Unidades	Coste (Ö)	Coste total (Ö)	Coste imputable
HP ProBook 4530s Inter® Core™ i5-2450M CPU 2.5GHz	1	730,00 "	730,00 "	85,05 "
Teclados Logitech X32	1	30,00 "	30,00 "	3,50 "
Ratón Logitech Mx518	1	25,00 "	25,00 "	2,91 "
Monitor TFT LG L2160HD	1	150,00 "	150,00 "	17,48 "
Impresora Epson	1	89,90 "	89,90 "	10,48 "
Pendrive Kingston 64 Gb	1	86,00 "	86,00 "	10,02 "
TOTAL				129,43 Ö

Tabla 60: Resumen de Hardware

Siendo la amortización:

$$\frac{C}{B} * \frac{A}{12} * D$$

A = nº de meses desde la fecha de facturación

B = periodo de depreciación (60 meses)

C = coste del equipo

D = % de uso que se dedica al proyecto

4.2.4 Resumen de Software y licencias

En la siguiente tabla se muestran las herramientas software necesarias para el proyecto. Todos los costes son calculados sin I.V.A.

Descripción	Unidades	Coste (Ö)	Coste total (Ö)	Coste imputable
Microsoft Windows 7	1	850,00 "	850,00 "	99,03 "
Microsoft Office Professional 2007	1	250,00 "	250,00 "	23,13 "
GanttProject	1	0,00 "	0,00 "	0,00 "
WebMatrix	1	0,00 "	0,00 "	0,00 "
Librería jQuery	1	0,00 "	0,00 "	0,00 "
Librería VISJS	1	0,00 "	0,00 "	0,00 "
Navegadores	1	0,00 "	0,00 "	0,00 "
TOTAL				122,16 Ö

Tabla 61: Resumen de Software y licencias

4.2.5 Resumen de material fungible

En la siguiente tabla se muestra el material fungible que se estima necesario para la realización del proyecto, así como sus costes. En material de escritorio englobamos folios, bolígrafos, gomas, lápices, clips, carpetas, archivadores, recambios, grapadoras y demás material de oficina. Todos los costes son calculados sin I.V.A.

Descripción	Coste total (€)
Material de escritorio variado	150,00 "
Recambios de impresora	100,00 "
TOTAL	250,00 €

Tabla 62: Resumen de material fungible

4.2.6 Resumen del presupuesto total

En la siguiente tabla se muestra el sumatorio de los totales anteriormente calculados pero sin el I.V.A. incluido. A la suma de los costes le vamos a añadir un diez por ciento en concepto de costes indirectos, lo cual equilibrará los riesgos del proyecto y aquellos otros valores que no se han tenido en cuenta al realizar el presupuesto.

Descripción	Coste total (€)
Equipo de trabajo	10.562,50 "
Subcontratación de tareas	0,00 "
Amortización Hardware	129,43 "
Amortización Software	122,16 "
Materiales Fungibles	250,00 "
Costes indirectos (21%)	2.323,46 "
TOTAL	13.387,55 €

Tabla 63: Resumen del presupuesto total

5 Conclusiones y Trabajos Futuros

5.1 Conclusiones

La principal motivación por la que realice este proyecto de fin carrera fue la posibilidad de realizar una herramienta que pudiera ser usada por un usuario final. Sin quedarse en un estudio, una simulación o un prototipo. Espero que la aplicación web final sea de utilidad tanto a mi tutor, origen de la idea, así como al resto del personal universitario.

Una vez concluido el proyecto de fin de carrera se puede proceder a analizar en profundidad lo conseguido durante las distintas fases de creación de la aplicación web.

El proyecto empezó con la idea de poder facilitar la localización de documentos científicos que pudieran ser interesantes para el usuario, por su relación con uno ya conocido. Para poder lograrlo se optó por una representación gráfica que siempre da una visión general de la información, permitiendo localizar rápidamente esos documentos que de forma contextual se podrían quedar perdidos. Además se buscó en todo momento que la aplicación web fuese sencilla e intuitiva.

Para la obtención de la información se decidió utilizar el portal de búsqueda de documentos CiteSeerX ya que dispone de una estructura muy tabulada que permitía una rápida extracción de la información, además de por el ámbito de los documentos que contiene, en su amplia mayoría sobre la ciencia de la computación.

Para la representación gráfica no se tuvo ninguna duda, dado que la relación entre los documentos no era jerárquica como sucede en una estructura de árbol y además se pueden crear ciclos entre los documentos el grafo fue la elegida.

Durante el desarrollo de la aplicación web se han encontrado diversos problemas. Siendo la obtención de los datos del buscador CiteSeerX la más importante. Este problema obligó a utilizar código en el lado del servidor (PHP) cuando se esperaba solo usar el lado del cliente (JavaScript), y así hacer una aplicación web independiente de la arquitectura del servidor de publicación. Otro de los problemas encontrados fue la creación del grafo y la limitación encontrada a la hora de pintar un gran número de nodos, pero se suplió con la parametrización del grafo, recortando ese posible gran número de nodos.

Se ha conseguido alcanzar de forma notable los objetivos fijados al inicio del proyecto de fin de carrera resolviendo todas las necesidades iniciales. Con la creación de la representación gráfica final en forma de grafo se ha cumplido los objetivos propuestos al principio de proyecto de fin de carrera.

El desarrollo de la aplicación web ha sido una experiencia enriquecedora ya que me ha permitido enfrentarme a complicaciones que hasta ahora no había encontrado en mi ámbito de desarrollo. Me ha permitido adquirir conocimientos en desarrollo de aplicaciones web al utilizar técnicas que hasta ahora no había usado.

5.2 Trabajos Futuros

Una vez concluido el proyecto de fin de carrera y aunque este ha cubierto los objetivos pedidos, se puede realizar ciertas mejoras además de ampliar o cambiar el ámbito de búsqueda. A continuación se van a citar las propuestas para trabajos futuros:

- Se ha utilizado el portal de búsqueda de CiteSeerX para obtener la información para la representación gráfica pero se puede optar por otro de los citados en la memoria, así se podrá ampliar el ámbito de búsqueda a otros tipos de documentos.
- Buscar una solución a la hora de obtener la información de la web de búsqueda evitando la ejecución de código en servidor.
- Para realizar el proceso de creación del grafo de forma más rápida se podría almacenar los datos del grafo en un fichero XML¹³ que pudiera ser usado en local sin necesidad de tener conexión a internet. Cuando se solicitará un documento se podría preguntar, si existe el fichero, si queremos usar el fichero local u obtener los datos del portal CiteSeerX de nuevo.
- Se podría crear una versión para móviles/tablet que haría más accesible la aplicación.
- Usar otra herramienta gráfica que permita un mayor número de nodos y aristas sin perder flexibilidad.

¹³ Sobre este punto ya esta creada la carga de un fichero con los datos del grafo que se uso durante la fase de desarrollo de la aplicación web.

6 Anexos

• Manual de Usuario

1. Instalación de la Aplicación Web (Publicación)

Para la instalación de la aplicación web se requiere de un servidor, con Windows o Linux, con un servidor web que interprete PHP, ejemplo Apache¹⁴, lighttpd¹⁵, Cherokee¹⁶ o IIS¹⁷. Que tenga salida a internet.

Se debe copiar la aplicación en la carpeta CiteGraph que se ubicará en la carpeta del servidor web donde se publiquen las web.

2. Ejecución de la Aplicación

Para ejecutar la aplicación se abrirá nuestro navegador preferido y se introducirá la dirección:

<http://nombre del servidor/CiteGraph/index.html>

3. Pantalla de Búsqueda de Documentos



Ilustración 16: Manual - Pantalla de Búsqueda

En la pantalla principal el usuario podrá realizar búsquedas de documentos. Introducimos el nombre del documento o parte de él y pulsamos ENTER o

¹⁴ <http://www.apache.org/>

¹⁵ <http://www.lighttpd.net/>

¹⁶ <http://cherokee-project.com/>

¹⁷ <https://www.iis.net/> - será necesario incluir el modulo de PHP

pinchamos en el texto **Buscar**. También podremos seleccionar el criterio de ordenación del resultado de la búsqueda.

How good is almost perfect Relevancia ▼ **Buscar**

Ilustración 17: Manual - Buscador

Una vez realizada la consulta y si esta tiene resultados, estos se visualizarán en páginas de 10 documentos donde se podrá el nombre del documento, un resumen del contenido del documento.

Universidad Carlos III de Madrid Visor Gráfico de Redes de Referencias Científicas

CiteSeer^x_{5M}

How good is almost perfect Relevancia Buscar

Resultado Búsqueda
Resultado 1 - 10 Total 1.118.773

[How good is almost perfect](#)
by Malte Helmert, Gabriele Röger - In ICAPS-Workshop on Heuristics for Domain-Independent Planning, 2007
"... Heuristic search using algorithms such as A* and IDA* is the prevalent method for obtaining optimal sequential solutions for classical planning tasks. Theoretical analyses of these classical search algorithms, such as the well-known results of Pohl, Gaschnig and Pearl, suggest that such heuristic ..."

[Perfect equilibrium in a bargaining model](#)
by Ariel Rubinstein - ECONOMETRICA, 1982
"..."

[Points-to Analysis in Almost Linear Time](#)
by Bjarne Steensgaard, 1996
"... We present an interprocedural flow-insensitive points-to analysis based on type inference methods with an almost linear time cost complexity. To our knowledge, this is the asymptotically fastest non-trivial interprocedural points-to analysis algorithm yet described. The algorithm is based on a non-s ..."

[Good features to track](#)
by Jianbo Shi, Carlo Tomasi, 1994
"... No feature-based vision system can work unless good features can be identified and tracked from frame to frame. Although tracking itself is by and large a solved problem, selecting features that can be tracked well and correspond to physical points in the world is still hard. We propose a feature se ..."

[Good Error-Correcting Codes based on Very Sparse Matrices](#)
by David J.C. MacKay, 1999
"... We study two families of error-correcting codes defined in terms of very sparse matrices. "MN" (MacKay-Neal) codes are recently invented, and "Gallager codes" were first investigated in 1962, but appear to have been largely forgotten, in spite of their excellent properties.

Ilustración 18: Manual - Resultado Búsqueda

En la parte superior de la lista de documentos resultantes se podrá visualizar el número total de documentos de la consulta, además del número de documento inicial y final dentro de la consulta.

Resultado Búsqueda
Resultado 1 - 10 Total 1.118.773

Ilustración 19: Manual - Resumen Búsqueda

Para avanzar o retroceder por las páginas de la consulta deberemos pulsar en el botón next o prev, según convenga.

← Prev 10 - Next 10 →

Ilustración 20: Manual - Navegación Búsqueda

Una vez localizado el documento del que queremos visualizar su grafo, pincharemos en el título. Esto hará que se abra una ventana con el grafo (ver punto 4).

4. Pantalla Grafo de Documentos

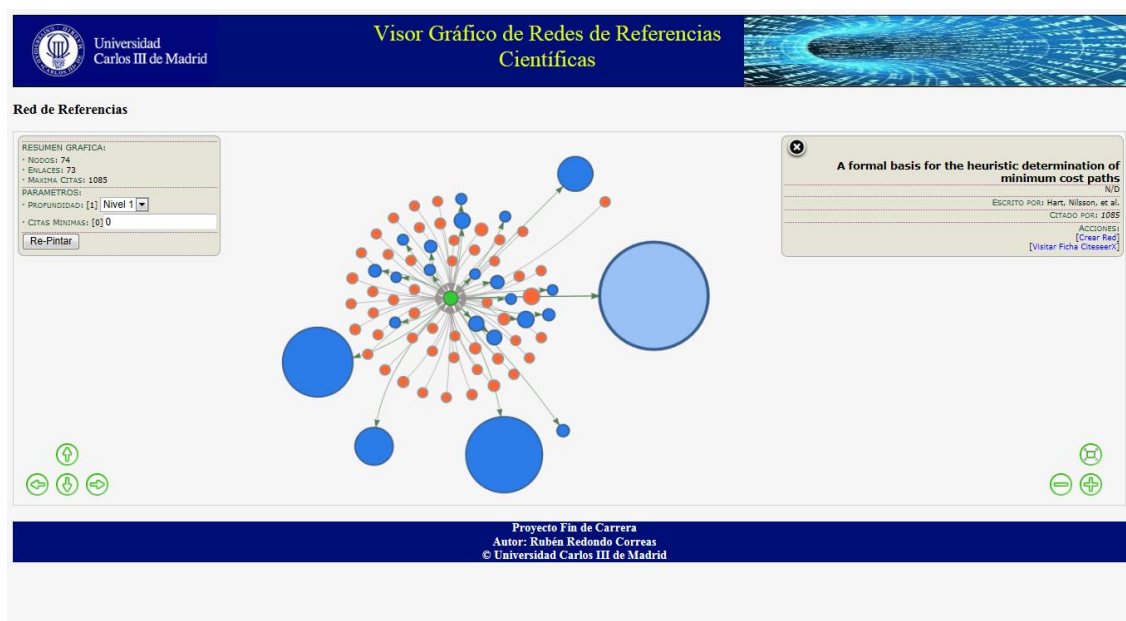


Ilustración 21: Manual - Visor Grafico

Una vez pinchamos en el título de uno de los documentos encontrados, se abrirá una nueva ventana donde se procesará a la obtención de los datos necesarios para la creación del dato, el tiempo de espera puede variar en función del número de documentos y la conexión con internet.

El grafo tiene las siguientes características:

- El tamaño que cada nodo representa la cantidad de documentos lo referencian.
- El color indica si el documento seleccionado (verde), un documento referenciado (azul) o un documento referenciador (rojo).

5. Manipulación del Grafo

Una vez se muestra el grafo, nosotros podremos manipularlo con las siguientes funciones:

- Mover el Grafo: Pulsa sobre el fondo del Grafo y arrástralo hasta posicionarlo donde se desee. También puede pulsar sobre los botones.



Ilustración 22: Manual - Navegación por el Grafo I

- Zoom: Pulsando sobre el botón y podemos hacer zoom al grafo.



Ilustración 23: Manual - Navegación por el Grafo II

- Mover un documento: Para cambiar la posición de un documento podemos arrastrarlo hasta la posición deseada.
- Nombre documento: Dejando el puntero del ratón sobre un documento se mostrará el nombre del documento así como el número de referencias.

6. Ficha del Documento

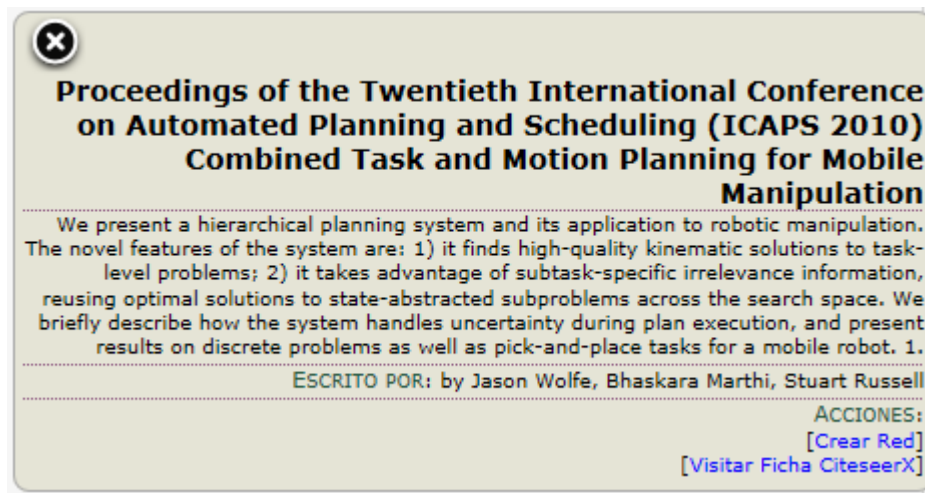


Ilustración 24: Manual - Ficha Documento

Para poder consultar la ficha de un documento deberemos pinchar sobre el documento. Aparecerá una ficha en la parte superior derecha del grafo. En ella se mostrará la siguiente información:

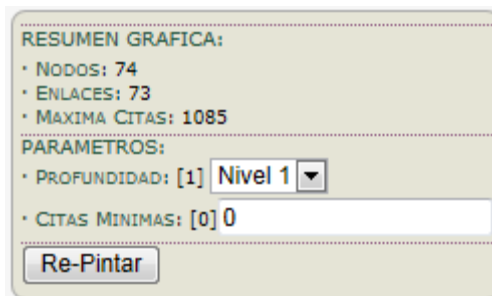
- Nombre
- Resumen
- Año
- Autores
- N^a Referencias

También podremos realizar las siguientes acciones:

- Generar el grafo de ese documento.
- Ver la ficha de la web de CiteSeerX

En cualquiera de las dos acciones la ventana no se cerrará.

7. Ficha Resumen Grafo



RESUMEN GRAFICA:

- NODOS: 74
- ENLACES: 73
- MAXIMA CITAS: 1085

PARAMETROS:

- PROFUNDIDAD: [1] Nivel 1 ▼
- CITAS MINIMAS: [0] 0

Re-Pintar

Ilustración 25: Manual - Ficha Resumen Grafo

Al terminar de cargar el Grafo se mostrará la ficha resumen en ella se podrá ver el número de nodos y aristas del grafo. Además se podrá manipular las características del grafo.

8. Aclaraciones

Para poder usar la aplicación web correctamente el servidor deberá estar conectado a internet.

Si la web de CiteSeerX estuviera caída o saturada es posible que las búsquedas y la obtención de los datos se paralicen dando un mensaje de error. La Universidad Carlos III de Madrid ni el desarrollador se hace responsable de la información obtenida de CiteSeerX y recuerda que el contenido y acceso a la web de CiteSeerX es público y gratuito.

- **Referencias**

LIBROS

- Jordi Collell Puig. *CSS· y JavaScript avanzado*. UOC
- Javier Eguíluz Pérez. *Introducción a JavaScript*. Librosweb
- Juan Diego Gauchat. *El gran libro de HTML5, CSS3 y JavaScript*. Marcombo
- Bear Bibeault, Yehuda Katz. *jQuery in Action*. Manning Publications

WEB

- <http://citeseerx.ist.psu.edu>
- <https://scholar.google.es/>
- <http://academic.research.microsoft.com>
- <https://www.mendeley.com/>
- <http://es.wikipedia.org/>

- http://www.w3schools.com/html/html5_intro.asp
- http://www.w3schools.com/css/css3_intro.asp
- <http://php.net/>
- <http://www.w3schools.com/js/>
- <http://jquery.com>
- <http://visjs.org>
- <http://es.wikipedia.org/wiki/Mendeley>

- **Acrónimos**

AJAX: *Asynchronous JavaScript And XML*

API: *Application Programming Interface*

CSS: *Cascading Style Sheets*

FTP: *File Transfer Protocol*

GPL: *General Public License*

HTML: *HyperText Markup Language*

HTTP: *HyperText Transfer Protocol*

PHP: *Hypertext Pre-Processor*

URL: *Uniform Resource Protocol*

W3C : *World Wide Web Consortium*

XML: *eXtensible*

UTF-8: *8-bit Unicode Transformation Format*

DOM: *Document Object Model*

PDF: *Portable Document Format*

XPath: *XML Path Language*

WWW: *World Wide Web*

DOI : *digital object identifier*